Computational Evolutionary Embryogeny

Or Yogev, Andrew A. Shapiro, *Senior Member, IEEE,* and Erik K. Antonsson, *Member, IEEE*

***Abstract*—Evolutionary and developmental processes are used to evolve the configurations of 3-D structures *in silico* to achieve desired performances. Natural systems utilize the combination of both evolution and development processes to produce remarkable performance and diversity. However, this approach has not yet been applied extensively to the design of continuous 3-D load-supporting structures. Beginning with a single artificial cell containing information analogous to a DNA sequence, a structure is grown according to the rules encoded in the sequence. Each artificial cell in the structure contains the same sequence of growth and development rules, and each artificial cell is an element in a finite element mesh representing the structure of the mature individual. Rule sequences are evolved over many generations through selection and survival of individuals in a population. Modularity and symmetry are visible in nearly every natural and engineered structure. An understanding of the evolution and expression of symmetry and modularity is emerging from recent biological research. Initial evidence of these attributes is present in the phenotypes that are developed from the artificial evolution, although neither characteristic is imposed nor selected-for directly. The computational evolutionary development approach presented here shows promise for synthesizing novel configurations of high-performance systems. The approach may advance the system design to a new paradigm, where current design strategies have difficulty producing useful solutions.**

***Index Terms*—Design synthesis, development, embryogeny, evolution, finite element, genetic algorithm, genome, modularity, morphogen, phenotype, structure.**

NOMENCLATURE

V_0	Initial volume of the cell.
h	Height of a selected cell face.
A_p	Area of a selected cell face.
R, S, T	Coordinates of a point inside a cell in local coordinates.
r, s, t	Coordinates of a point inside a cell in local coordinates.
$\mathbf{F}$	Deformation tensor.
$\mathbf{r}$	Vector form of r, s, t.
$\mathbf{R}$	Vector form of R, S, T.
α	Scalar which represents change in volume.
v	Cell volume after a geometric operation is applied.
V	Cell volume before a geometric operation is applied.
$\mathbf{J}$	Jacobian matrix.
N_i	Interpolation function.
X_i, Y_i, Z_i	Coordinates of node i in the reference configuration.
$d\mathbf{R}, d\mathbf{S}, d\mathbf{T}$	Infinitesimal material vectors in the X, Y, and Z directions, respectively.
$\mathbf{a}$	Directional vector originating in the reference configuration.
κ_1	First coefficient for the shearing operation.
κ_2	Second coefficient for the shearing operation.
$\alpha_1, \alpha_2, \alpha_3$	Expansion coefficients.
σ	Cauchy–Green tensor.
s	Morphogen diffusion rate.
d	Morphogen diffusion constant.
E_c	Energy consumption of the cth cell.
B_c	Metabolic rate of the cth cell.
S	Total volume of the phenotype.
N_c	Number of cells comprising the phenotype.
k	Node number.
$x_m^{(k)}$	Vector coordinate of the kth node.
m	Dummy index.
α_m	Deformation measure of a node, used as part of the repair process.
x_m	Node coordinates in the present configuration, used as part of the repair process.
f_0	Total deformation level of the phenotype.
g_m	Degree of deformation of the cell.
$e_{m,k}$	Vector notation, used in the process of computing f_0.
$x_{m,k}$	Node coordinates, used in the process of computing f_0.
x	Node coordinates.
J_m	Vector, used in the process of computing f_0.
μ_i	Preference function for the ith variable.
S_i	ith performance variable.
a,b	Preference function parameters.
α	Slope parameter for preference functions.
ω_i	Importance weight for aggregation.
$\mathcal{P}_s$	Aggregation function.
s	Degree of compensation for aggregation.

Manuscript received February 8, 2008; revised March 8, 2009. First version published October 30, 2009; current version published March 31, 2010.

O. Yogev is with eSolar, Inc., Pasadena, CA 91103 USA, and also with the Engineering Design Research Laboratory, Division of Engineering and Applied Science, California Institute of Technology, Pasadena, CA 91125 USA (e-mail: yogevori@gmail.com).

A. A. Shapiro is with the Department of Electrical Engineering and Computer Science, University of California, Irvine, CA 92697 USA, and also with the Enterprise Engineering Division of the Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA 91109 USA (e-mail: andrew.a.shapiro@jpl.nasa.gov).

E. K. Antonsson is with the Division of Engineering and Applied Science, California Institute of Technology, Pasadena, CA 91125 USA (e-mail: erik.antonsson@caltech.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TEVC.2009.2030438

I. Introduction

NATURAL EVOLUTION has produced systems of fantastic complexity, robustness and adaptability. Recent research has shown that it is the combination of both evolution and development processes that has produced these remarkable results [1], [2]. Evolution does not act directly on the configurations of adult phenotypes, rather it successively alters and revises the *rules* that guide the growth of a zygote into an embryo and its further development into an adult.

The computational evolutionary development approach presented here is based on an artificial evolution using indirect encoding of growth and development rules. The approach was tested on a classical structural engineering problem. The results demonstrate that artificial evolution and embryogenesis can synthesize phenotypes with novel configurations that exhibit modularity, and which meet performance goals.

The objective of this research is not simply to evolve structures that meet the desired performance requirements, but also to explore complex design environments (including functionally-graded, tailored composite materials) and to establish a fundamental understanding of the origin and development of *modularity* in design, and the design rules that give rise to modular design configurations.

A. Definitions

Phenotypes are:

> *the set of observable characteristics of an individual resulting from the interaction of its genotype with the environment.*[1]

Rules, in this context as in nature, are encoded in the genome of each individual. Genes regulate the production of proteins using transcription factors, and hence the growth and development of the organism. The genome contains the set of instructions for the development process while the environment provides inputs that regulate the instructions [3].

Natural evolutionary processes refine the sets of rules, in the form of genes, which result in adult forms. Natural selection acts upon the phenotypes, thus rewarding sets of rules that produce fit individuals. The *indirect* character of the encoding of genetic information in natural systems, and the inter-relationship between evolution of rules and the growth and development of adult forms, has been responsible for the diversity, complexity, modularity, robustness, and adaptability in the natural world [4], [5].

Embryogeny or *Embryogenesis* is the process of growth by which a genotype develops into a phenotype, and is central to the emerging understanding of the relationship between evolution and development.[2]

B. Prior Related Work

Artificial evolution, in the form of genetic algorithms (GAs), has been used in a wide variety of application areas [7]–[10]. The goal of this prior work has been to address problems where solutions can be identified that exhibit one or more improved features. The key element of GAs, first stated by Holland [11], is *implicit parallelism*. The idea is that the scheme of actions employed in a GA (selection, cross-over, and mutation performed on N individuals) implicitly searches for an optimum in N^3 space. This result is powerful, since it enables the rapid exploration of large solution spaces.

The majority of optimization problems that make use of the genetic algorithm approach have employed *direct* encoding. When using direct encoding, there is a one-to-one relationship between the genetic information in an individual and the configuration of the individual. Most commonly, the genetic information contains a description of the individual, in contrast with indirect encoding, where the genetic information contains a set of rules that, when executed (and perhaps influenced by various environmental factors), guide the growth and development of a single cell into an adult.

Genetic algorithms have been previously applied as an optimization method in structural evolution. Traditional structural evolutionary methods generally start by generating a fixed mesh grid (like a chess board) with a predefined volume and constraints, such as external forces and boundary conditions [12]. Every cell in the grid can have one of two states; either material is present or absent. The genetic information in this approach contains information indicating the material state in each cell of the grid. Once the initial configuration and the boundary conditions are defined (loads, constraints, etc.), an evolutionary process searches for the configuration exhibiting the best performance. This is an evolution using a *direct* encoding, with no embryogeny, and thus there is a one-to-one mapping between the genotype and phenotype, resulting in an optimization problem with a large, but finite, number of possible states [13]. This method produces structures that reflect the underlying shape of the grid [7] and is not able to create continuous or smooth structures. These grid-based building-block structures are not only unrealistic from an engineering perspective, but from a mathematical point of view they search in a limited solution space, resulting in local optima.

Indirect encoding, and the study of artificial embryogeny, has been proposed previously [6], [14]–[16]. In most examples, the goal has been to grow and evolve a predefined target shape on a predefined grid starting from a single cell, using simple rules such as cell division and protein diffusion. Early work in this area has demonstrated the ability of indirect encoding to produce modular phenotypes in graphs and patterns [17]–[19].

The study of computational embryogeny has previously been conducted using a variety of approaches. The field of autonomous agents is one where this technique first made an impact, specifically in the area of neural networks. The notion of utilizing a collection of simple basic elements (autonomous agents) to solve a complex problem has proven to be highly efficient [20]. Vaario [21] has built an adaptive neural network that grows and adapts with respect to an external environment such that every neuron is an autonomous agent. He used an L-system concept to build a stimulus tracking system [22]. In his system, he defined agents as

[1] Oxford American Dictionary.

[2] "It should be noted that the correct term is *embryogeny*, which refers to the process, rather than the oft-misused term *embryology*, which refers to the science of studying embryos and embryogenies." [6]

sets of neurons with basic rules for growth and development regulated by the environment. Once these rules were evolved, artificial organisms were created that were capable of adapting themselves to changes in the environment. Vaario was also one of the pioneers who understood the importance of the role of the environment in solving complex problems artificially [23]. Kitano [24] first used a context-free grammar, in the form of the L-system, to represent a neural network. He showed that this indirect representation speeds up the rate of convergence gradually. Following this work, he developed an artificial growth model with basic metabolic rules [25]. He showed that after hundreds of generations, these rules can produce complex networks that are self organized. Following Kitano's work, Eggenberger introduced an artificial evolutionary and development system [12]. In his system cells were created with a set of genes that were to be evolved into a desired 3-D shape in response to the concentration field of a morphogen [15], [17], [26]–[29].

The work presented here builds on these ideas of growth and development, and demonstrates the unconstrained evolution of rules that produce structures comprising multiple materials, where fitness is determined only by structural performance.

II. Approach

In the work reported here, an artificial embryogeny has been created for structures. The two critical foundational elements of the work are the selection of the artificial cell (the basic structural element), a collection of which constitutes each individual, and the artificial genes (the rules) which are evolved into the genetic information for each individual.

The genetic information of an individual is shared by all of its cells. Each individual cell executes its rules until a mature structure is formed. Once maturity is reached, an evaluation scheme determines the fitness (performance) of the structure. Evolutionary operations (selection, crossover, and mutation) alter and refine the genetic information in a population of individuals over multiple generations. The results are structures that meet the desired performance goals.

A. Material Properties

One of the main advantages of the approach presented here is the ability to evolve inhomogeneous structures with a high degree of internal complexity. It has been observed that the material properties of biological structures are unique and complex. A wide range of stiffnesses and strengths are exhibited, and in many cases a combination of a high degree of compliance with high strength produces robust structures that are difficult or impossible to replicate with the engineered materials of today. One contributing factor to this difference is inhomogeneity: the material properties of many natural structures change from location to location.

Engineered materials with similar degrees of inhomogeneity are just now beginning to be available, through techniques such as shape deposition manufacturing (SDM) [30]. SDM creates structures by adding and binding materials with different mechanical properties droplet by droplet. Design approaches to best take advantage of these new tailor-able inhomogeneous engineered materials have not yet been developed.

III. Evolutionary and Developmental Scheme

The evolutionary and developmental scheme used here is derived from a genetic algorithm, and is illustrated in Fig. 1. The growth and development process in particular is shown in Fig. 2.

The algorithm is initialized with sets of randomly generated genomes. Development begins with a single artificial cell. This cell is placed on (and attached to) a ground plane. A load vector, to be supported by the evolved and developed structure, is positioned above the initial cell. The origin of the load vector radiates a signal, analogous to a morphogen. The goal is for individuals comprising one or more cells to grow to reach the height of the morphogen, and to be able to support the load.

One individual is grown from each genome, by executing the rules in the genome. Once each individual reaches maturity, its fitness is evaluated by means of a finite element analysis and additional parameters. The selection process is based on the *roulette wheel* method, where the fitness values of each individual are used to select parents to produce offspring, where a better fitness value results in a higher probability of being selected for reproduction.

Once two parents have been selected, they produce offspring through a crossover process. In this process, the genome from each parent is cut at a randomly selected word boundary, where a word is a sequence of rules. One portion of the gene string from each parent is joined together producing a child. The remaining gene strings from each parent are joined together producing a second child.

Similar to evolution in nature, the genome is also subject to random mutation. The mutation process can erase an entire word and replace it with another, or replace a single rule within a word.

These three steps—selection, crossover, and mutation—are repeated, and each repetition is defined as one generation.

Due to the difficulty in evolving structures that grow to the desired height and support the load, a staged evolutionary scheme is used. This approach is similar to incremental problem complexity [31], where task complexity grows with the developing ability of evolved phenotypes. In the application of this scheme here, phenotypes are gradually challenged with increasingly difficult tasks as the evolution proceeds, i.e., the heights of both the morphogen and the load to be supported are increased.

IV. Cells

The model begins with a structural element that is analogous to a biological cell. Cells in nature act as building blocks for organisms [32]. The basic structural element in the approach presented here is an extended 3-D triclinic hexahedral finite element "brick," illustrated in Fig. 3. Each cell-like finite element contains an identical copy of the genetic information

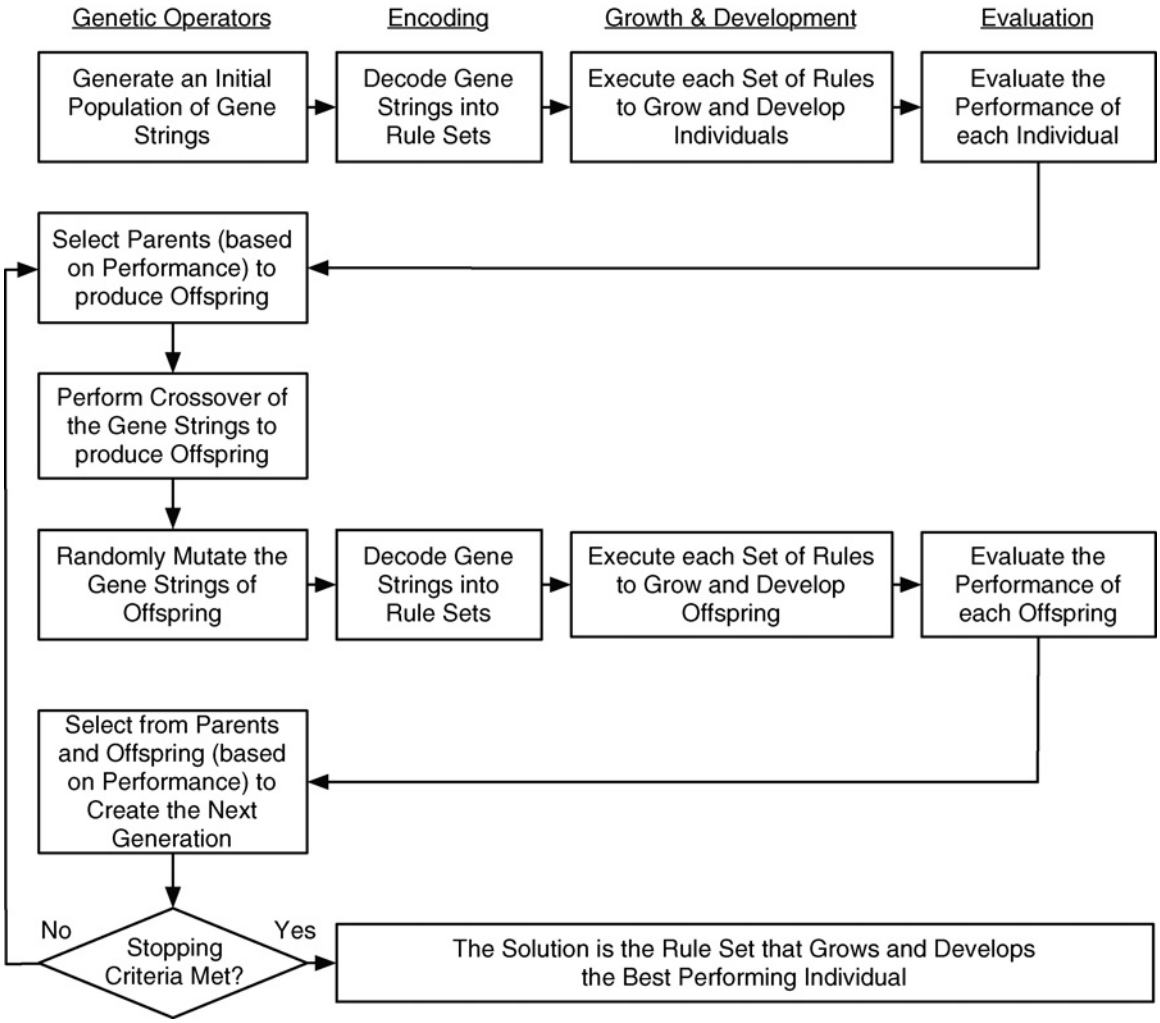


Fig. 1. Computational evolutionary development process.

of the individual. A collection of cells forms a finite element mesh, which defines the phenotype.

V. Development Process

In the development process a single cell grows and develops into a mature adult phenotype. The process starts with a single cell placed on the ground subjected to a gravity field. The cell is initially made of steel. A morphogen radiating from the position of the load is placed at a location above the ground plane. The morphogen diffusion and the gravity field create an environment that is sufficient to regulate the rules in the genome. The execution of rules modifies existing cells and produces new cells that contain an identical copy of the genome. The new cells are also subjected to the global environment (gravity and the morphogen concentration) and their local environment (stresses and signals from nearby cells). Because the environment of each cell is different, the regulation mechanisms may cause different rules in different cells to be executed. This repeated process of cell production and rule/gene regulation creates a phenotype which will eventually grow and reach the load morphogen. Once a phenotype has reached the load, the load is applied to the top cells that coincide with the morphogen. This load generates a mechanical stress distribution along the phenotype which will alter the local environment on the cells, and may cause the rule/gene regulation mechanisms to alter the further growth and development of the cells. The process of evaluating the mechanical stresses on the cells at each time step is performed by solving a finite element scheme.

VI. Genes/Rules

Rules are the basic instructions encoded inside the genome. Here, rules dictate the growth and development processes of each cell of the phenotype. Mimicking nature, the basic structure of genetic information is an *if*-conditional *then*-action rule.

Two main principles guided the creation of the rule elements (conditionals and actions) to form genes. The first principle was to create conditional rules that respond only to the local environment of the cell. The second principle was to choose action rules that will not put any constraint on the topology of the phenotype, and can generally develop any 3-D shape.

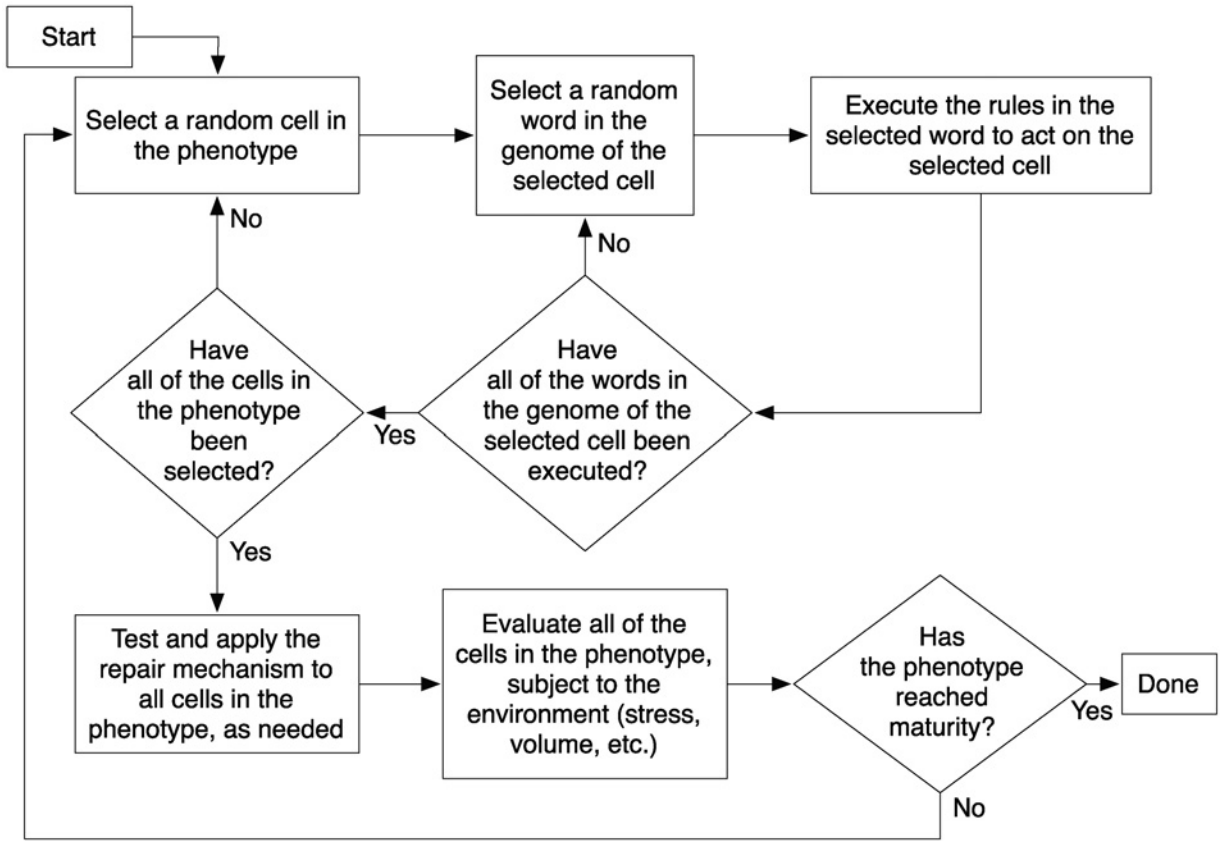


Fig. 2. Growth and development process.

In addition to these two principles, rule elements were added that are observed in nature, such as cell differentiation and veto rules. All of these rules play a significant role in the development of phenotypes.

The rules used here fall into three groups: geometric, cell-type actions, and conditionals in the form of veto tests. Geometric rules represent instructions which have an effect on the shape of the cell. Cell-type rules alter the type of the cell, and therefore mimic the basic process observed in cell-biology. Veto tests affect other instructions at the genome level and will be discussed first.

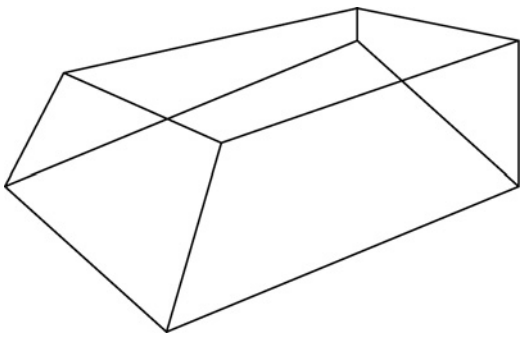


Fig. 3. Basic structural hexahedral "brick" 3-D finite element.

A. Conditionals

Conditional artificial rules are "veto" or "suppression" rules. These rules affect other rules only at the genome level, by turning actions off or on according to whether the conditional test is satisfied or not. Veto genes that switch regulatory mechanisms on or off have been observed in biology [1]. For example, tumor suppression genes that turn off other genes that produce tumor cells [33].

B. Cell-Type Actions

Cell-type rules are actions which mimic basic operations in cell-biology. Four basic operations are used here: cell division, cell differentiation, cell death, and cell adhesion. Each of these operations was modeled as a single rule.

1) *Cell Division:* The cell division rule is responsible for creating new mass and thus generates more building blocks (new cells) for the construction of the phenotype. Cells divide with respect to a given vector in the reference configuration of the cell. The vector specifies a face of the new cell. The selection of the face uses the inverse iso-parametric mapping [34], and is performed by translating the vector into the cell center point and then determining the face which intersects the translated vector. Once the face has been determined, a new triclinic (hexahedral) cell is created perpendicular to the selected face, such that the total volume of both cells equals the volume of the parent cell prior to division, as shown in Fig. 4. V_0 is defined as the initial volume of the cell prior to division. An isotropic shrink operation is

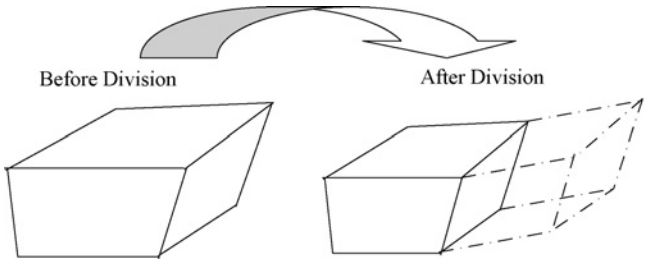


Fig. 4. Cell division operation.

applied with parameter $\alpha = \sqrt[3]{\frac{1}{2}}$. This operation will shrink the cell into half of its original volume. The new cell is an extension by height h of the selected parent face. The face area is identified by A_p, and then the extension height of the new cell is determined as

$$h = \frac{V_0}{2A_p}. \tag{1}$$

A cell will attempt to divide, prioritized according to the parameters governing the rules in its genome.

2) *Cell Differentiation:* Two different kinds of cells are modeled, one made of steel and the other made of aluminum. The cell differentiation rule simply alters the material properties of a cell from one type to the other.

3) *Cell Death:* Cell death is a self extermination mechanism, which kills the cell and removes it from the phenotype, and from further consideration in the embryogeny computations.

4) *Cell Adhesion:* During growth, two cells may intersect, causing them to adhere. In nature, this fundamental process creates the integrated structures forming the configuration of the organism. Studies have shown that the speed of the adhesion process is short [35]. Adhesion of nearby cells is modeled in a manner similar to the adherence of bubbles [36].

When two cells intersect, the merging procedure deforms the cells by applying a set of displacement fields to the nodes corresponding to the faces to be merged. The displacement fields must be physically admissible from a continuum mechanics standpoint, and result in a minimum distortion for both cells, which thus minimizes the total strain energy of the two adhering cells.

Three major steps are executed during the cell adhesion process: first, determine the corresponding faces to be adhered; second, determine the pairs of nodes to be merged, and third, optimize the positions of the merged nodes, as shown in Fig. 5.

An overview of the adhesion process can be found in Appendix B.

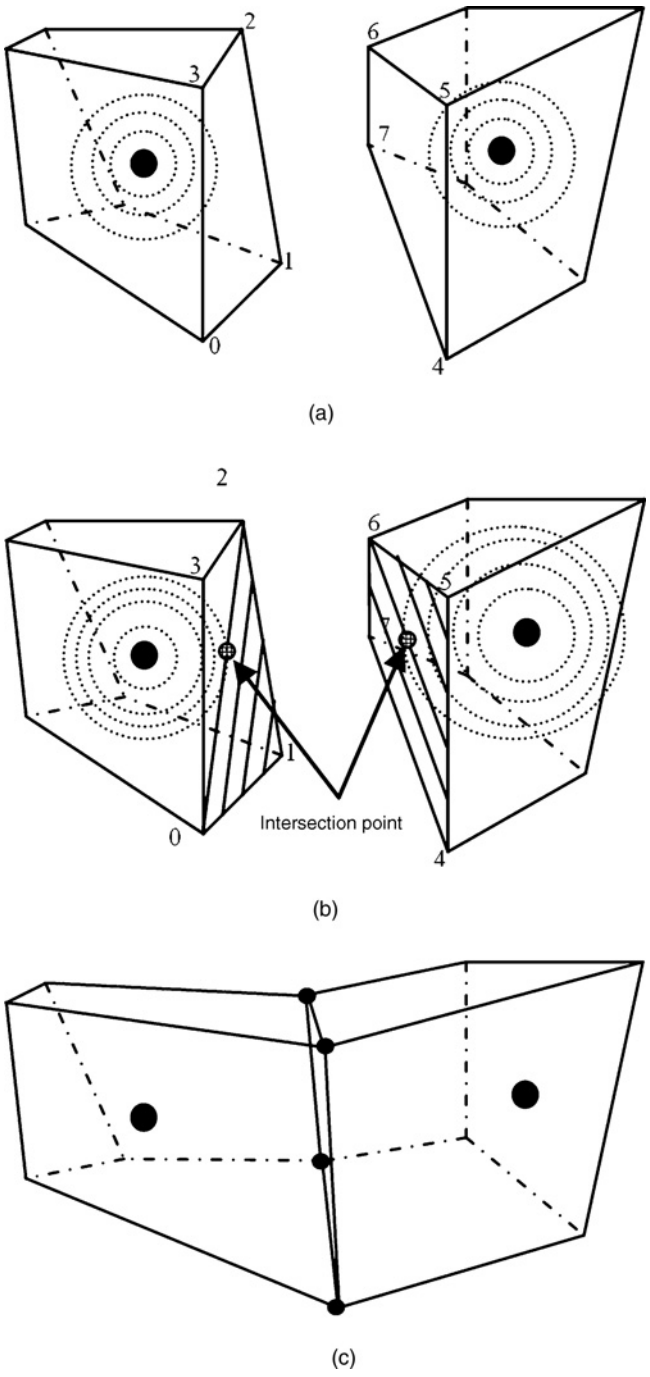


Fig. 5. (a) Two cells prior to adhesion with radiating signals from the center of each cell. (b) Faces of the two cells to be adhered. (c) Cells after adhesion.

C. Geometric Actions

The idea of formalizing the geometric rules arises from studies of the development processes of plants. Plants are remarkable engineering structures that sustain high dynamic loads, for example, those generated by wind. The topological structure of plants and their material complexity provide them with the ability to sustain high mechanical stresses. During the natural embryogenesis of plants, as in the model presented here, every 3-D region (which is a collection of several cells) can deform according to nine different geometric operations: one for isotropic growth, two for anisotropic growth, three for shear, and three for rotation [37], illustrated in Fig. 6. Every cell in the model presented here represents a 3-D region that can be deformed according to these geometric operations. Every geometric operation, excluding rotation, corresponds to a geometric rule.

These rules are the basic tools to generate any topological structure excluding arches and round objects. Nevertheless, even arcs or curved objects can be represented by linear

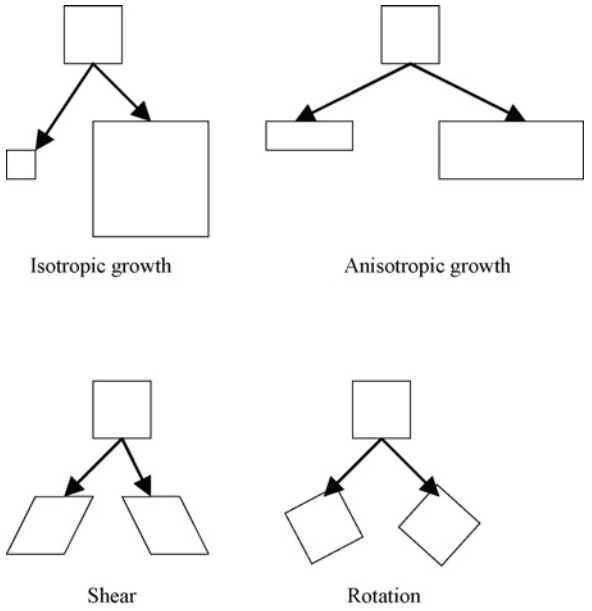


Fig. 6. Four basic geometric operations observed in sub-regions of plants.

elements with high accuracy, as shown in Fig. 7(a). The last claim fails to hold for a fixed grid, as shown in Fig. 7(b). Even a simple arc has a crude representation in the fixed mesh-grid approach.

The action of a geometric rule is always applied in the reference configuration. Since there is a one-to-one mapping between the local and the reference configuration, it will be more convenient to execute the geometric operations locally and then map the result back to the reference configuration. In order to be consistent with the next derivations, the faces of the cell in the local configuration have been numbered according to Fig. 8.

1) *Growth:* The action of this rule is to expand the cell by an amount in all directions. Assuming the cell is in the local configuration, as shown in Fig. 3, the following mapping is defined by (2). Every point R, S, T, corresponding to a point inside the cell before applying the geometric rule, is mapped to a point r, s, t after the geometric rule has been applied. Both points are defined in the coordinate system of the local configuration

$$r = \alpha_1 R \ ; \ s = \alpha_2 S \ ; \ t = \alpha_3 T \tag{2}$$

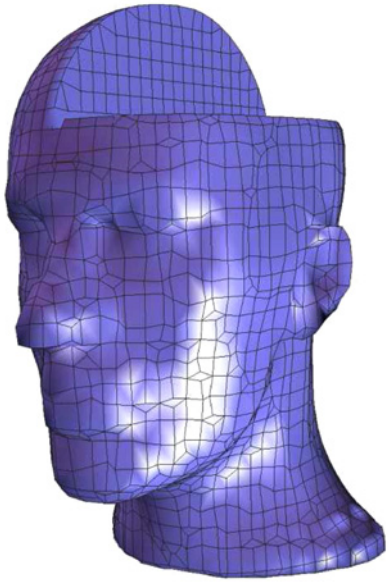
where $\alpha_1, \alpha_2, \alpha_3$ are coefficients representing the expansions along each of the three orthogonal axes. When $\alpha_1 = \alpha_2 = \alpha_3$, the growth is isotropic.

The deformation gradient tensor for this mapping is defined in

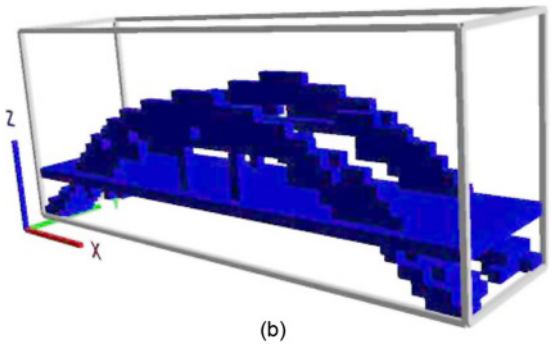
$$\mathbf{F} = \frac{\partial \mathbf{r}}{\partial \mathbf{R}} = \begin{bmatrix} \alpha_1 & & \\ & \alpha_2 & \\ & & \alpha_3 \end{bmatrix}. \tag{3}$$

The volume change of any infinitesimal point dV to dv can be computed according to

$$\frac{dv}{dV} = \det(F) = \alpha_1 \alpha_2 \alpha_3. \tag{4}$$



(a)



(b)

Fig. 7. (a) 3-D hexahedral linearly conformal mesh [38]. (b) 3-D rectilinear grid mesh [39].

Using (4), the volume change of the entire cell can be computed according to

$$\frac{v_{\text{cell}}}{V_{\text{cell}}} = \mathbf{J} = \det(F) = \alpha_1 \alpha_2 \alpha_3. \tag{5}$$

The mapping in (2) is applied to the nodes of the cell in the local configuration. Since the mapping is homogeneous, mapping the nodes will set the map of the entire cell. By setting α and applying this mapping, the change in volume of a cell in the local configuration will be $\alpha_1 \alpha_2 \alpha_3$. Define $\tilde{V}$ to be the volume of the cell in the reference configuration prior to this mapping. A unit volume[3] of the cell in its reference configuration prior to the application of this mapping is given in (6). The quantities $\mathbf{R}, \mathbf{S}, \mathbf{T}$ are defined as material fibers in the local configuration

$$d\tilde{V} = N_i(d\mathbf{R})\, X_i \times N_i(d\mathbf{S})\, Y_i \times N_i(d\mathbf{T})\, Z_i. \tag{6}$$

[3]A cube with volume = 1 in the local configuration.

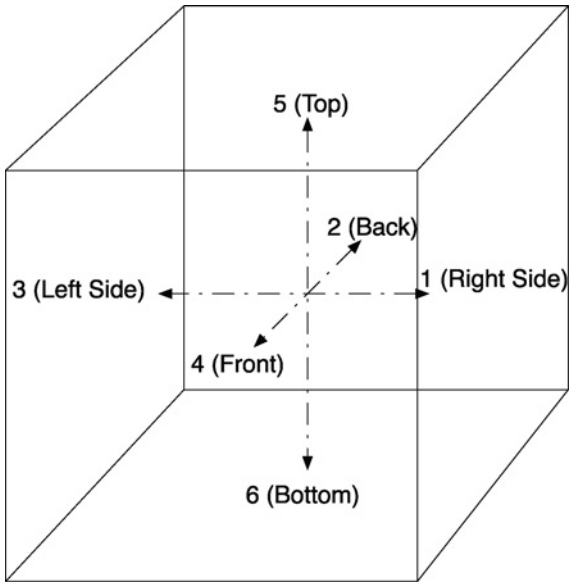


Fig. 8. Face numbering in the local configuration.

A unit volume of the cell in its reference configuration, after this mapping has been applied, is given in (7), where $\mathbf{r}, \mathbf{s}, \mathbf{t}$ are material fibers in the local configuration after the mapping has been applied

$$d\tilde{v} = N_i\,(d\mathbf{r})\,X_i \times N_i\,(d\mathbf{s})\,Y_i \times N_i\,(d\mathbf{t})\,Z_i. \tag{7}$$

Again using the definition of the gradient deformation tensor $\mathbf{F}$ shown in (3), (7) can be written in a new form as

$$d\tilde{v} = N_i\,(\mathbf{F}d\mathbf{R})\,X_i \times N_i\,(\mathbf{F}d\mathbf{S})\,Y_i \times N_i\,(\mathbf{F}d\mathbf{T})\,Z_i. \tag{8}$$

Since N_i is a linear operator, (8) can be rewritten into the form

$$\begin{aligned} d\tilde{v} &= \mathbf{F}N_i\,(d\mathbf{R})\,X_i \times \mathbf{F}N_i\,(d\mathbf{S})\,Y_i \times \mathbf{F}N_i\,(d\mathbf{T})\,Z_i \\ &= \mathbf{J}N_i\,(d\mathbf{R})\,X_i \times N_i\,(d\mathbf{S})\,Y_i \times N_i\,(d\mathbf{T})\,Z_i = \mathbf{J}d\tilde{V}. \end{aligned} \tag{9}$$

The volume ratio of the cells in the reference configuration, after the homogeneous mapping (2) has been applied, is equal to the volume ratio of the cells in the local configuration, subjected to the same mapping. This conclusion enables the growth operation to be applied in the local configuration of the cell.

2) *Shear:* Under the execution of a shear rule, a cell in the reference configuration will be deformed in a given direction such that its original volume will be preserved. The shearing operations involve two steps: given a directional vector $\mathbf{a}$ originating in the reference configuration, the first step is to determine the face of the cell which first coincides with an extension of this vector. There are many ways to perform this test; the inverse isoparametric mapping method is used here. First, the direction vector is translated to the center of the cell. Next, a point along this vector interior to the cell is chosen. By mapping the coordinates of this point to the local configuration, the inverse mapping of the vector is simply the difference between the coordinates of the inverted points, and

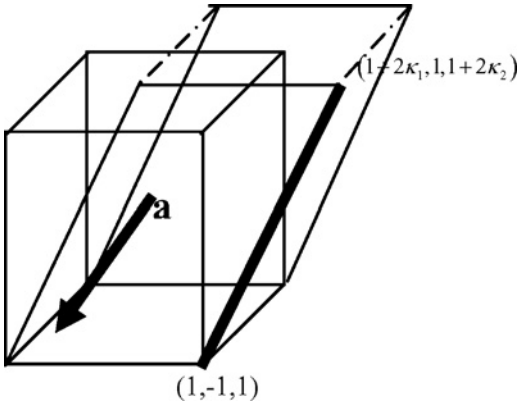


Fig. 9. Shear operation.

the center of the cell in the local configuration system, which is $(0, 0, 0)$. The face of the cell in the local configuration coinciding with the inverted vector corresponds to the term with the maximum value of the inverted vector. The steps to compute the inverse isoparametric mapping can be found in Appendix A. The second step is to deform the cell in the local configuration such that the edges which are perpendicular to the selected face become parallel to $\mathbf{a}$, as shown in Fig. 9. The mapping, defined in (10) shears the cell in a predetermined direction such that the volume of the cell remains the same. The number below each box in (10) corresponds to a particular face with respect to the notation defined in Fig. 8

$$\begin{array}{ccc} \begin{array}{l} r = R + \kappa_1(S+1) \\ s = S \\ t = T + \kappa_2(S+1) \end{array} & \begin{array}{l} r = R + \kappa_1(T-1) \\ s = S + \kappa_2(T-1) \\ t = T \end{array} & \begin{array}{l} r = R \\ s = S + \kappa_1(S-1) \\ t = T + \kappa_2(S-1) \end{array} \\ 1 & 2 & 3 \\ \begin{array}{l} r = R + \kappa_1(T+1) \\ s = S + \kappa_2(T+1) \\ t = T \end{array} & \begin{array}{l} r = R + \kappa_1(S+1) \\ s = S \\ t = T + \kappa_2(S+1) \end{array} & \begin{array}{l} r = R + \kappa_1(S-1) \\ s = S \\ t = T + \kappa_2(S-1) \end{array} \\ 4 & 5 & 6 \end{array}. \tag{10}$$

It is easy to show that each of the mappings in (10) is isochoric (volume preserving), by following the same derivation as in (3), (4), and (5).

The relation between the coefficients κ_1, κ_2 and the target direction vector $\mathbf{a} = \{a_1, a_2, a_3\}$ is shown in (11). As before, each face corresponds to a different relation between $\mathbf{a}$ and κ

$$\begin{array}{ccc} \begin{array}{l} \kappa_1 = \frac{a_2}{\sqrt{1-a_2^2-a_3^2}} \\ \kappa_2 = \frac{a_3}{\sqrt{1-a_2^2-a_3^2}} \end{array} & \begin{array}{l} \kappa_1 = \frac{-a_1}{\sqrt{1-a_1^2-a_2^2}} \\ \kappa_2 = \frac{-a_2}{\sqrt{1-a_1^2-a_2^2}} \end{array} & \begin{array}{l} \kappa_1 = \frac{-a_2}{\sqrt{1-a_2^2-a_3^2}} \\ \kappa_2 = \frac{-a_3}{\sqrt{1-a_2^2-a_3^2}} \end{array} \\ 1 & 2 & 3 \\ \begin{array}{l} \kappa_1 = \frac{a_1}{\sqrt{1-a_1^2-a_2^2}} \\ \kappa_2 = \frac{a_2}{\sqrt{1-a_1^2-a_2^2}} \end{array} & \begin{array}{l} \kappa_1 = \frac{a_1}{\sqrt{1-a_1^2-a_3^2}} \\ \kappa_2 = \frac{a_3}{\sqrt{1-a_1^2-a_3^2}} \end{array} & \begin{array}{l} \kappa_1 = \frac{-a_1}{\sqrt{1-a_1^2-a_3^2}} \\ \kappa_2 = \frac{-a_3}{\sqrt{1-a_1^2-a_3^2}} \end{array} \\ 4 & 5 & 6 \end{array}. \tag{11}$$

VII. Environment

The environment in which the individuals are grown contains factors which every cell can sense, and which can affect the way rules are expressed. The environment can trigger the execution of rules and control their expression. Biologists studying growth and development have found that the concentration of morphogens [40] and mechanical stresses [41] are two crucial factors which influence the growth of a phenotype. Various mathematical models illuminate the role the two effects play in determining the size and shape of tissues [42], [43]. In natural systems, information from the environment is transferred to the cells through proteins known as receptors. The receptors transfer the information by generating chemicals that diffuse through the cell membrane at some level of concentration. This concentration stimulates the action of rules at the local cell level during the developmental stage.

In the simulated evolutionary embryogenesis here, as well as in nature, the relationship between the information that cells receive from the environment and the development of the phenotype is not predetermined. Rather, conditionals are available to the evolutionary process that sense the concentration or gradient of each morphogen. In this way, the evolutionary process establishes the relationship between information and growth and development.

The morphogens represent points, surfaces or volumes in space, with associated engineering requirements. In the artificial embryogeny presented here, two kinds of morphogens are present. The first morphogen represents an external load to be supported by the phenotype. The morphogen is located at a predetermined point and produces a chemical which continuously diffuses in space according to (12). The parameter s represents the concentration of the morphogen, while d represents the distance from the location of the morphogen. The morphogen diffuses through space impinging on the walls of each cell. The second morphogen represents the surface of the ground to which cells adhere when they intersect the surface

$$s = e^{-d}. \tag{12}$$

The phenotype is subjected to gravity effects and an external load. These two effects generate internal mechanical stresses within the cells, which can be represented by the Cauchy–Green tensor given in

$$\sigma = \begin{bmatrix} \sigma_{xx} & \sigma_{xy} & \sigma_{xz} \\ \sigma_{xy} & \sigma_{yy} & \sigma_{yz} \\ \sigma_{xz} & \sigma_{yz} & \sigma_{zz} \end{bmatrix}. \tag{13}$$

By solving a finite element scheme at every time step, the mechanical stress distribution within the cells can be calculated [34]. There are six independent parameters which can be derived from the Cauchy stress tensor and added as environmental factors. The parameters are: the three principal values and the three principal directions of the Cauchy stress tensor. Since every point inside the cell has a different stress tensor, only the point which has the maximum principal stress is considered. The calculations are outlined in Appendix A.

Two additional environmental factors correspond to the volume and the age of the cell. The volume of a cell is simply the value of the Jacobian, defined in (A.7) in Appendix A. The age of the cell corresponds to the number of time steps that have passed since the cell was created. Cells also maintain information about their distance from neighboring cells. This information is used to trigger the execution of the cell adhesion rule.

VIII. Genome Structure

The genome contains words which contain rules with their corresponding letters (Tables I–IV). A word is simply a sequence of rules. The letter "Z" indicates the beginning of a set of rules within a word. A veto rule can only act on the remaining rules within a word. After every generation, a search routine looks for identical words within the same genome. These words are combined together with the letter "R" indicating the number of times the particular word will be executed in one time frame. The length of the genome can vary between different individuals, but has a predefined maximum length.

A. Syntax Rules

Every rule comprises one capital letter and several lower case letters, each with a fractional coefficient. The capital letter identifies the type of a rule. The lower case letters correspond to the environmental factors which guide the action of the rule within the cell. The fractional coefficient is a number between 0 and 100 which represents the level of expression of the rule with respect to the environmental factor that follows it (a mechanism analogous to transcription factors in nature). Tables I–III contain four columns, the first column corresponds to the identifying letter, and the second column corresponds to the name of the rule. The third column specifies how many additional parameters each rule has while the fourth column specifies the type of additional parameters, as listed in Table IV.

The rule execution process begins by reading the identifying letter of the rule, followed by a set of environmental factors each with its fractional coefficient. Each environmental factor is normalized to a nominal value. The following normalizations are used here: the mechanical stress is normalized to the yield stress of the material, the volume of the cell is normalized to the initial volume of the first cell, the intensity of the morphogen is normalized with the morphogen intensity impinging on the first cell at the initiation of the growth process, at the first time step, and the age of the cell is normalized with the maximum age allowed. The product of the environmental factor and the fractional coefficient specifies the level of expression of the rule. For instance, the set of letters S30g reduces the volume of the cell by 30%.

In a similar way, the word "R1ZC10i" corresponds to: R1, repeat once; Z, word boundary indicator; C10i, cause the cell to grow isotropically by 10% based on the load morphogen concentration that was measured by that cell.

TABLE I
GEOMETRIC RULES

ID	Name	N	Optional Parameters
A	Shear	1	(d, e, f, h) × fractional coefficient
B	Anisotropic growth	3	(a, b, c, g, i) × fractional coefficient
C	Isotropic growth	1	(a, b, c, g, i) × fractional coefficient
S	Isotropic shrink	1	(a, b, c, g, i) × fractional coefficient

N = number of parameters

TABLE II
CELL-TYPE OPERATIONS

ID	Name	N	Optional Parameters
D	Cell division	U	(d, e, f, h)
K	Cell death	0	
F	Cell differentiation	0	

N = number of parameters
U = unlimited number

TABLE III
VETO (CONDITIONAL) OPERATIONS

ID	Name	N	Optional Parameters
V	Suppress below	1	(a, b, c, g, i) × fractional coefficient
W	Suppress above	1	(a, b, c, g, i) × fractional coefficient

N = number of parameters

IX. METABOLISM AND THERMODYNAMICS

A thermodynamic energy model which balances the energy required to maintain the organism mass with the energy required to create a new mass [44] is incorporated in the growth and development process used here. The amount of energy E_c that each cell consumes in a given time step is proportional to its metabolic rate B_c. Part of this energy is used for maintaining the existing phenotype while the remaining energy may be used for creating new mass, as shown in

$$E_c = E_0 B_c \Delta t. \tag{14}$$

TABLE IV
CELL INFORMATION

ID	Description
a	Maximum principal stress normalized with the yield stress
b	Middle principal stress normalized with the yield stress
c	Minimum principal stress normalized with the yield stress
d	Principal vector corresponding to the maximum principal stress
e	Principal vector corresponding to the middle principal stress
f	Principal vector corresponding to the minimum principal stress
g	Cell size
h	Load morphogen intensity
i	Load morphogen direction
t	Cell age

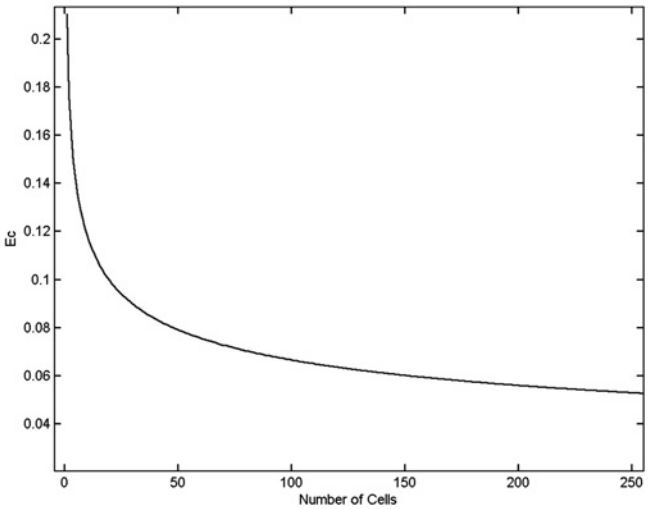


Fig. 10. Amount of energy E_c available to each cell during growth, as a function of the total number of cells in the growing phenotype.

Using Kleiber's Law [45], [46], and assuming small variation in the volume of the cells,[4] the metabolic rate B_c of each cell is proportional to the size of the phenotype S (the total volume of the phenotype) divided by the number of cells, N_c, as shown in (15). The parameter E_0 is a proportionality constant which sets the energy scale

$$B_c \propto \frac{S^{3/4}}{N_c}. \tag{15}$$

By combining (14) and (15), and by setting $E_0 = 1$, a thermodynamic size limit can be specified for the phenotypes, as shown in (16). At the beginning of every time step, each cell contains an amount of energy E_c. This energy is utilized by the cell to execute its genome. Every rule (operation) may consume a different amount of energy

$$E_c = \frac{S^{3/4}}{N_c}. \tag{16}$$

The following scheme addresses the process of assigning energies to rules. First, consider a scenario where there is nonstop equally sized cell production. Assuming $E_0 = 1$, E_c can be plotted, as shown in Fig. 10. The total amount of energy, E_c, is reduced as the number of cells increases since the volume of the phenotype increases. Setting the energy consumption of the cell-division rule to be above $E_c(N)$ forces a weak upper limit to the size of the phenotype. All the other energies are set to be less than the division energy. This enables the phenotype to change its topology without adding new mass.

The advantage of using this approach is that there is no predefined upper bound, or other limit, on the size of the phenotype. Even when the phenotype reaches the thermodynamic limit, this approach will permit new mass to be created at the expense of removing existing mass. This potentially changes the topology of the phenotype. However, the thermodynamic balance will not prevent phenomena such as unlimited cell

[4]Small variation of the volume of cells is a requirement in the fitness function.

division or extermination of the entire phenotype. These last phenomena are addressed by evolution and disease mechanisms.

X. Time Increments

The growth and development processes in nature proceed continuously. In order to simulate these processes numerically, a time step has been defined. A time step starts when the first cell executes its first rule and ends when the last cell executes its last rule.

Between the end of a time step and the beginning of the next time step, three processes are executed. The first process is an iterative process which repairs damaged cells using two optimization schemes, discussed later in Section XI-A. The second process is an evaluation of the phenotype by computing the mechanical stress distribution across the cells using a finite element scheme. In the third process, the internal energy of the cell is updated according to (16). In this discretization time scheme, rules are always executed in the reference configuration. In other words, even though an execution of a rule might change the topological representation of a phenotype and thus effect the stresses inside the cells, these changes will only take place in the next time step.

XI. Diseases

Individuals may suffer from a disease during growth and development. A disease only occurs as a consequence of a defective genome, and diseases adversely affect the growth process of the phenotype. When a developmental disease is detected in an individual, a penalty is applied to the measure of its performance. The following diseases are present.

1) *High Cell Division Rate:* an upper limit to the number of divisions at one time step is set. If the number of cell divisions exceeds the threshold value, it is identified as a disease.
2) *Self Extermination:* A defective genome might contain instructions that will eliminate all of the cells in that phenotype at some point during its development. This is identified as a disease.
3) *Morphology of the Cell:* As part of the repair process (which is a subprocess of the growth and development process illustrated in Fig. 2) every cell is regularly tested with (17) and (19) to determine whether it is "tangled" or highly deformed. An automatic repair mechanism (which is described in the following section) attempts to fix any cell that exceeds threshold values. If the repair mechanism fails to repair the cell, then the growth process is stopped. The phenotype is then evaluated, and a performance measure is assigned to the phenotype based on its current state. Since the phenotype fails to reach maturity, the performance measure is penalized (reduced by a factor). This process produces a comparative benefit for individuals that reach maturity, but will not eliminate phenotypes that have good properties but fail to reach maturity.

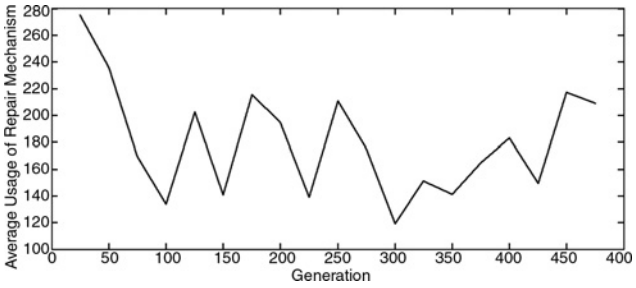


Fig. 11. Average usage of the repair mechanism during evolution.

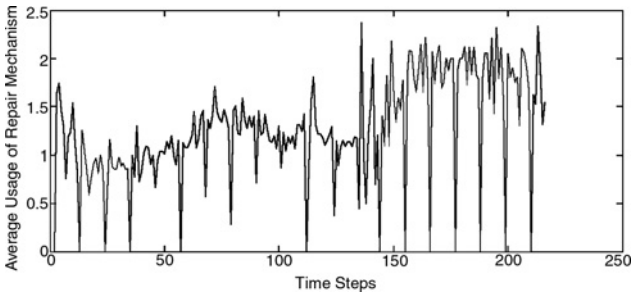


Fig. 12. Average usage of the repair mechanism with respect to time for one phenotype during the growth and development process.

A. Repair Process

Each phenotype is a collection of cells which is also a finite element mesh. During the growth and development process, cells may be deformed significantly due to the execution of rules (especially geometric operations). In order to accurately evaluate the mechanical stress distribution, the shape of the cells must obey two restrictions: convexity and deformation. The convexity restriction states that every cell has to be convex, which is determined by computing the Jacobian in (A.7) (in Appendix A) and checking that it is positive. If the Jacobian is negative then one or more nodes are identified as being "tangled." A repair process will then "untangle" these nodes, as described below. Fig. 11 shows the average usage of the repair mechanism during the process of evolution. The average usage is defined as the total number of times the repair mechanism is used by all cells through all of the time steps for the phenotype to reach maturity, divided by the sum of the number of cells in the phenotype at each time step. The use of the repair mechanism generally decreases in the beginning evolutionary process. Fig. 12 shows the utilization of the repair mechanism during the growth process of one phenotype. The y-axis represents the average number of times each cell is repaired during the development process. On average, 1.27 cells were repaired at each time step.

Following the work of Peter Knupp [47]–[51], α_m is defined as a scalar according to (17), where $x_m^{(1)}$, $x_m^{(2)}$, $x_m^{(3)}$ correspond to the coordinates of three nodes which are adjacent to node $x_m^{(0)}$, where m is a dummy index, and a negative value of α_m corresponds to a tangled node which needs to be untangled. The untangling process is performed using the conjugate gradient method with a global function that is to be minimized defined in (18). Because a tangled finite element mesh cannot be evaluated, the repair process, which is described detail in [50],

is necessary for the evolutionary development presented here

$$\alpha_m = \left(x_m^{(1)} - x_m^{(0)}\right) \times \left(x_m^{(2)} - x_m^{(0)}\right) \times \left(x_m^{(3)} - x_m^{(0)}\right) \tag{17}$$

$$f_0 = \frac{1}{2}\sum_{m=1}^{M} \{|\alpha_m| - \alpha_m\}. \tag{18}$$

The second restriction corresponds to deformation level. A conditional number, g_m, is defined in (19), where $x_{m,k}$ are the coordinates of nodes adjacent to node x, $g_m \geq 1$

$$\begin{aligned} e_{m,k} &= x_{m,k} - x \\ J_m &= \left[e_{m,1}, e_{m,2}, e_{m,3}\right] \\ g_m &= \det\left|J_m^T J_m\right|. \end{aligned} \tag{19}$$

The value of g_m represents the degree of deformation of the cell. A high conditional number corresponds to a highly deformed cell that needs repair. The repairing mechanism is to minimize the objective function f_0 defined in (20), where α_m is defined in (17). Further details may be found in [49]

$$f_0 = |J_m|\,|Adj\,(J_m)| \,/\, |\alpha_m|\,. \tag{20}$$

Each of the repair processes is an iterative optimization process. If either process fails to converge after a large number of iterations, a disease will be identified.

XII. Maturity

Once the growth process becomes stable, the phenotype is identified as being mature. The maturity stage is defined when no new mass is created or removed from the phenotype in a predefined time frame. Determining the length of this time frame is tricky since a phenotype might reach a limited stable region and then may continue to grow. In the method presented here, the following heuristic was utilized: the stability time frame was taken to be twice the previously observed period of stable size during the development of that particular individual. The initial period of stability is taken to be 20 time frames without adding or deleting cells, and with a change in volume of the phenotype of no more than 5%.

Fig. 13 shows a typical growth process of a representative phenotype. The y-axis marks the number of cells in the phenotype while the x-axis represents the development time. During the initial stage of the growth process, the production rate of new cells is relatively high. As time evolves, this rate decreases and stabilizes. There are two stabilization regions (plateaus) in which no new mass is created. The first plateau is unstable since the phenotype continues to produce new mass after some time. The second plateau is stable. The phenotype has reached stability such that no additional mass is created.

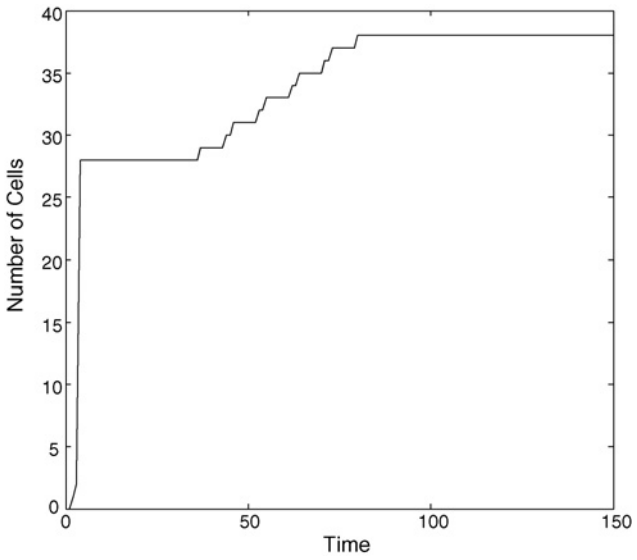


Fig. 13. Increase in the number of cells in a phenotype during a typical development process.

XIII. Fitness Evaluation

A. Fitness Description

Each phenotype is evaluated with respect to six performance attributes: mechanical stress, weight, shape of the cells, distance from the load point, age of the phenotype, and the maximum volume of the largest cell in the phenotype. The mechanical stress corresponds to the maximum value of the von Mises stress in the phenotype. The weight corresponds to the weight of the entire phenotype. The shape of the cells corresponds to the average value of the conditional number of all the nodes in the phenotype, defined in (19). The distance to the load point corresponds to the minimum distance that exists between any one of the cells to the load point. If one of the cells intersects with the location of the source of a morphogen then this distance is zero. The age of the phenotype corresponds to the number of time steps a particular individual has been growing without developing a disease. The parameter that establishes the maximum volume of a single cell puts constraints on the size of the individual cells, preventing extremely large cells in the phenotype.

1) *Aggregation:* In order to establish the fitness of individuals in the population, multiple performance values are aggregated into a single scalar [52], [53]. The objective is to minimize the scalar fitness value.

The aggregation approach builds on the prior engineering design work of Scott and Antonsson [54], where both importance weighting and degree of compensation among the variables are utilized. The degree of compensation specifies how a strong value of a particular performance variable may compensate for a deficiency of another variable.

To begin, the value of each performance variable is mapped to a preference value between 0 and 1 by a preference function, where a preference value of 1 corresponds to a perfectly acceptable value of the performance variable; a preference of 0 corresponds to a completely unacceptable value of the performance variable. A preference function μ_i, maps every variable S_i to a value on the real interval line $[0, 1]$ using (21), as illustrated in Fig. 14. The variable a represents the value such that any value below it will be considered to have a preference value of 1. The variable b represents the maximum value such that any value above will be considered

to have a preference value of 0. Setting a and b controls the range of feasible solutions and also provides the ability to put constraints on the phenotype.

The slope of the function specifies the improvement rate which corresponds to a particular variable. One of the major concerns which must be taken into account in choosing the preference function corresponds to the volatility of the individuals within the population. A single mutation in the genome may turn an individual with high performance into a non-feasible one. While these phenomena provide the ability to explore a large span of the space of possible solutions, its drawback is that it significantly penalizes individuals with poor performance. These phenomena of poor behavior are often observed at the early stages of the evolutionary process where most of the population has low measures of performance.

The solution implemented here is first to identify the particular variables in the fitness functions that have high volatility and produce evaluations of poor performance. These variables will have a preference mapping in the form of an exponentially decaying function, shown in (22), where α sets the slope. By reducing the value of α, the preference corresponding to the variable b is increased. The value of b can be increased such that a wider range of values will be considered and thus, individuals with varying degrees of poor performance can be distinguished from one another

$$\mu_i : S_i \rightarrow [0, 1] \tag{21}$$

$$\mu_i = \begin{Bmatrix} 1 & S_i < a \\ e^{-\alpha(S_i - m)} & a \leq S_i < b \\ \varepsilon & S_i \geq b \end{Bmatrix}. \tag{22}$$

Once all the preference functions have been defined, the weight and the degree of compensation of these values are used to compute the aggregated measure of performance $\mathcal{P}_s$ in (23). The weight for the ith performance variable is represented by ω_i, and s represents the degree of compensation. Observing (23), it can be seen [54] that $s = 1$ corresponds to a weighted sum, the $\lim s \rightarrow 0$ corresponds to the weighted product $\left(\mu_1^{\omega_1} \mu_2^{\omega_2} \ldots \mu_n^{\omega_n}\right)^{\frac{1}{\omega_1+\omega_2+\ldots+\omega_n}}$, and $\lim s \rightarrow -\infty$ corresponds to $\min(\mu_1, \mu_2, \ldots, \mu_n)$. Here, only compensation values $s \leq 0$ are used

$$\mathcal{P}_s\left((\mu_1, \omega_1), (\mu_2, \omega_2), \ldots, (\mu_n, \omega_n)\right) = \left(\frac{\omega_1 \mu_1^s + \omega_2 \mu_2^s + \ldots + \omega_n \mu_n^s}{\omega_1 + \omega_2 + \ldots + \omega_n}\right)^{\frac{1}{s}}. \tag{23}$$

XIV. Example

The approach described above has been applied to problems similar to ones observed in engineering and nature. The configuration of a structure to be synthesized is one that is capable of supporting a highly variable load 7 m above the ground. In addition, the structure is to be as light-weight as possible.

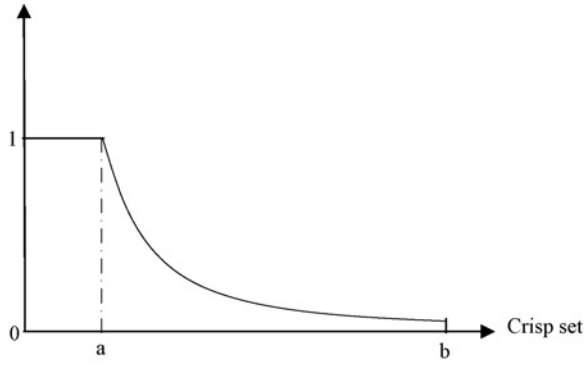


Fig. 14. Representative preference function, where the abscissa is values of a performance variable, and the ordinate is preference, where a preference value of 1 corresponds to a perfectly acceptable performance value and 0 corresponds to an unacceptable performance value.

TABLE V
Fitness Function Parameters

ID	Value
s	-1.0
w_{Stress}	1.0
w_{Mass}	0.1
w_{Shape}	0.1
w_{Distance}	1.0
w_{Age}	0.5
$w_{\text{MaximumVolume}}$	1.0

In order to provide a basis for comparison, a simple solution has been constructed to serve as a reference, illustrated in Fig. 15. The solution is a straight vertical beam with a 1 m × 1 m cross section. The solution in Fig. 15 corresponds to a fitness value of 6.2054, by using (23), with the weights w and the degree of compensation s set according to Table V.

Equation (23) provides an aggregate measure of performance $\mathcal{P}_s$ between [0, 1], such that better performance results in a higher aggregate performance measure. The inverse of (23) is used here, shown in (24) for fitness $\mathcal{F}_s$, such that the goal of the evolution is to minimize the aggregated fitness

$$\mathcal{F}_s\left((\mu_1, \omega_1), (\mu_2, \omega_2), \ldots, (\mu_n, \omega_n)\right) = \frac{1}{\left(\frac{\omega_1 \mu_1^s + \omega_2 \mu_2^s + \ldots + \omega_n \mu_n^s}{\omega_1 + \omega_2 + \ldots + \omega_n}\right)^{\frac{1}{s}}}. \tag{24}$$

Nature has evolved phenotypes that address this problem in different ways. Trees, for instance, in addition to other functions, support loads generated by the their own structure and the wind. Bones support gravity and muscle loads.

As described above, two morphogens are present in the environment: one represents the load to be supported; the other represents the ground, illustrated in Fig. 16. To introduce variation into the environment, the direction of the load changes at each time step during development of the individual phenotype, as shown by the multiple load vectors in Fig. 16 and the pink sector in Figs. 19–23 and 28. Prior work has shown that environmental changes can promote robustness [55] and modularity [19].

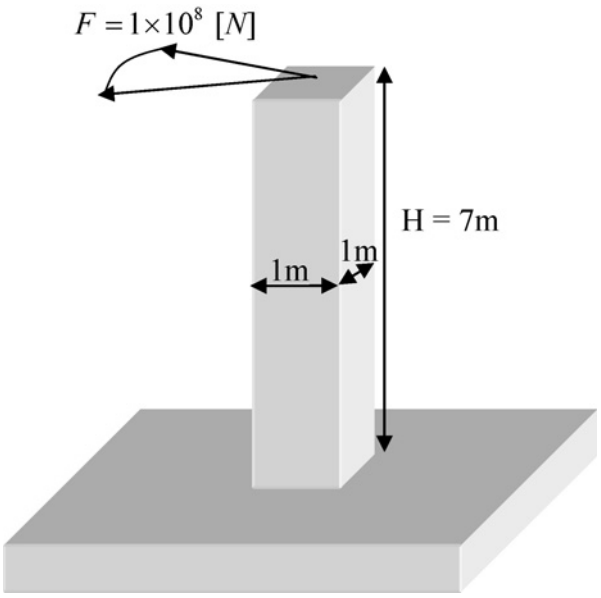


Fig. 15. Simple reference solution.

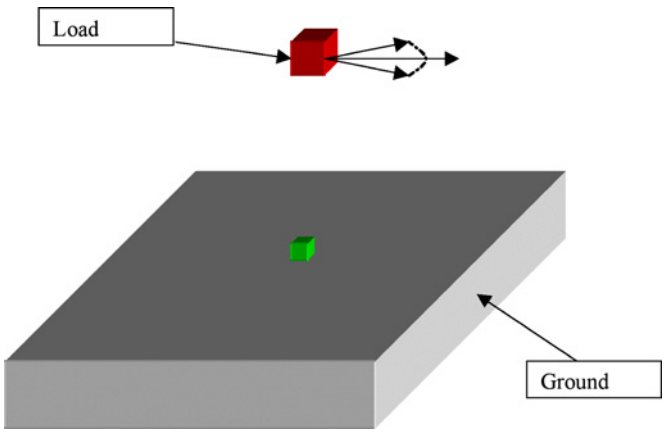


Fig. 16. (color online) Gray box represents the ground. The small green cube is the initial cell of an individual. The red cube represents the load. The load is 20 000 N and its direction is randomly selected in the range ±45° at each time step during development of the individual phenotype. The final distance between the ground and the load is 10 m.

As mentioned previously, the height of load morphogen is periodically increased during the evolutionary process.

XV. Results

The simulated evolution was run in parallel on 70 processors for 24 h. The initial population contained 400 individuals, each one starting with a random genome. Fig. 17 shows the fitness value of the best phenotype in the population with respect to the number of generations, the vertical dotted lines correspond to times during the evolution when the height of the load was increased. It can be seen that the fitness value oscillates during the evolutionary process. These phenomena correspond to the increasing increments of the height of the load. After every increase, it takes several generations to evolve the phenotypes with respect to the new location of the load and thus to minimize fitness.

Fig. 18 shows the fitness evaluation of the best phenotype once the load has reached the desired height of 7 m. The horizontal dotted line corresponds to the fitness value of the reference solution. The evolved phenotype is approximately three times better in performance than the reference solution.

The fitness values in Fig. 18 exhibit small oscillations. These phenomena have been observed many times during the evolutionary process and is a consequence of the variability in the environment. This kind of variability can result in different phenotypes grown from the same genotype through the regulation of genes/rules during the growth process.

2) *Phenotype Analysis:* A typical developmental process of an evolved phenotype is shown in Fig. 19. The development begins with a single cell and two morphogens [Fig. 19(a)]. The load is simulated by a vector force that randomly changes direction within the pink sector. Prior to the phenotype growing to reach the final height of the load, it is only subjected to gravity [Fig. 19(b)]. Once the phenotype grows to reach the final height of the source of the load morphogen, the phenotype is subjected to the full load. The colors of the cells represent mechanical stress. Green corresponds to low stresses (below the yield stress), and red corresponds to high stresses (above the yield stress) [Fig. 19(c)].

The topology is similar to a pyramid, but unfortunately, almost all of the cells in the phenotype are over-stressed (and are therefore colored red) indicating that this phenotype does not perform well in supporting the load (fitness = 102.54). As a result, this genome is unlikely to be selected for crossover, and therefore its genetic information is likely to be eliminated from the evolution.

Figs. 20 and 21 show two phenotypes that have been evolved after 400 generations. Fig. 20 shows two different views of one phenotype and Fig. 21 contains two views of a second. Both phenotypes are able to support the load, indicated by the color of the cells. However, the configurations of the two are different. The first phenotype is composed of two major modules (structural elements) that are perpendicular to each other. One of the modules is primarily vertical, while the other extends diagonally, which increases the ability of this phenotype to support the varying load. The two modules are connected to each other by means of cells that transfer the load from one module to the other, shown in Fig. 20(b). The second phenotype has the configuration of a double helix [Fig. 21(a) and (b)]. This phenotype contains two primary modules (distinct structural elements) that are wrapped together and connected at multiple locations. The helix has a unique structural topology which makes it able to support loads that vary in direction.

Characteristics of modularity (i.e., distinct structural elements) can be observed in the evolved phenotypes. At this stage of the research the degree modularity of each phenotype cannot be quantified, and therefore no measure is calculated or reported. However, both of the modular characteristics shown here have spontaneously emerged without directly imposing them in the fitness function or constraining the configuration of the phenotype.

3) *Development Analysis:* Fig. 22 illustrates the embryogenesis of the helically-shaped phenotype shown in Fig. 21.

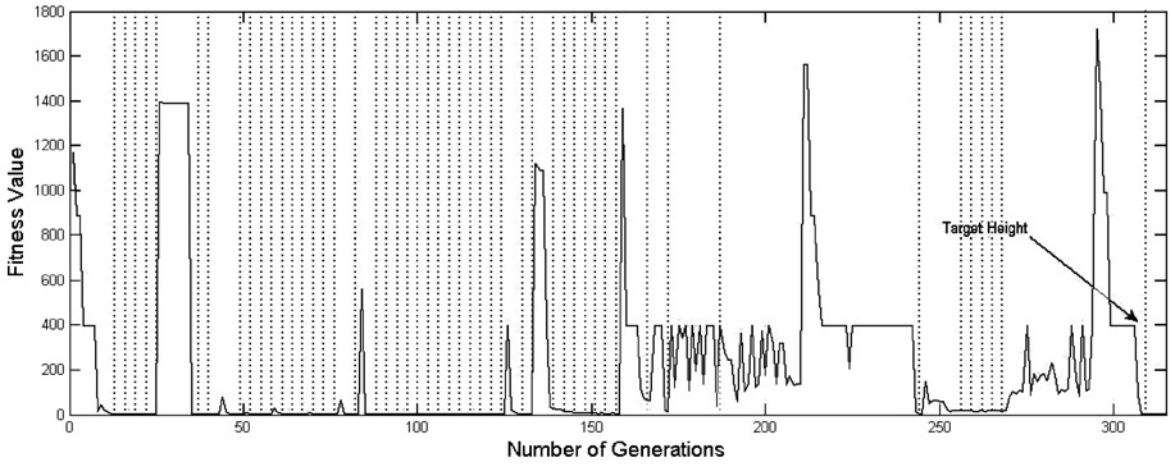


Fig. 17. Evolution of fitness values with respect to the number of generations. The vertical dotted lines show when the height of the load was increased.

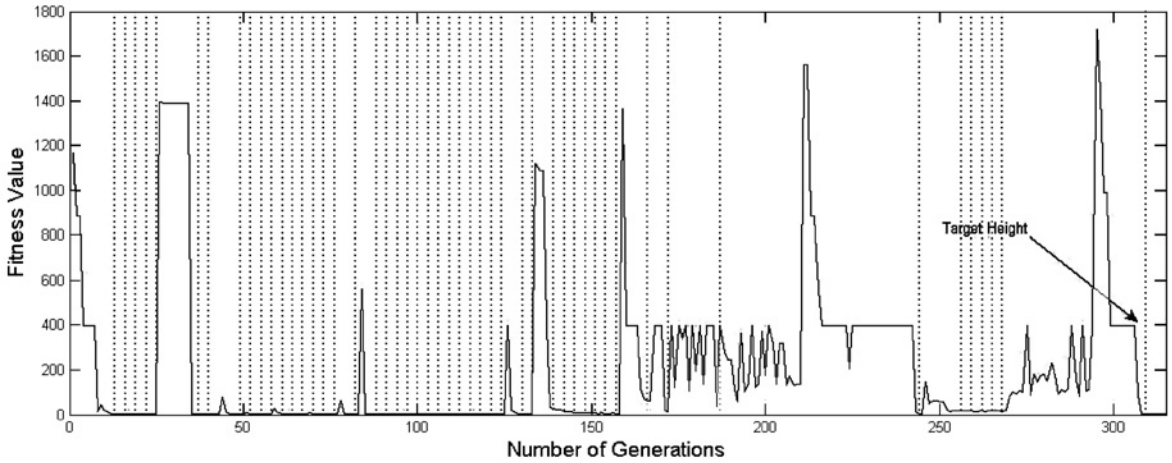


Fig. 18. Evolution of fitness values once the phenotype has reached the desired height.

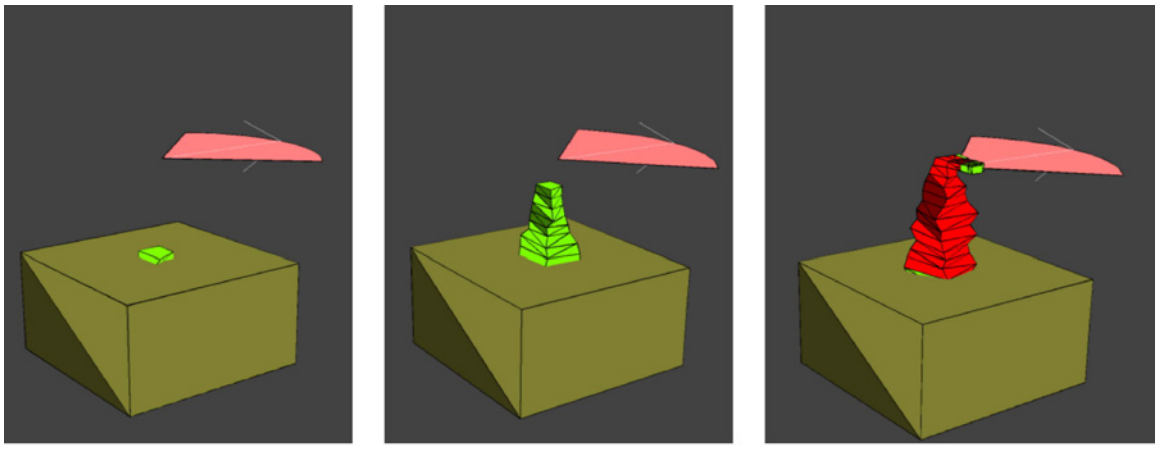


Fig. 19. (color online) Three stages during the developmental process of a single evolved phenotype. The colors represent mechanical stress. Green corresponds to low stresses (below the yield stress), and red corresponds to high stresses (above the yield stress). The white arrow represents the magnitude and principal direction of the load. The pink sector represents the range of variation of the direction of the load.

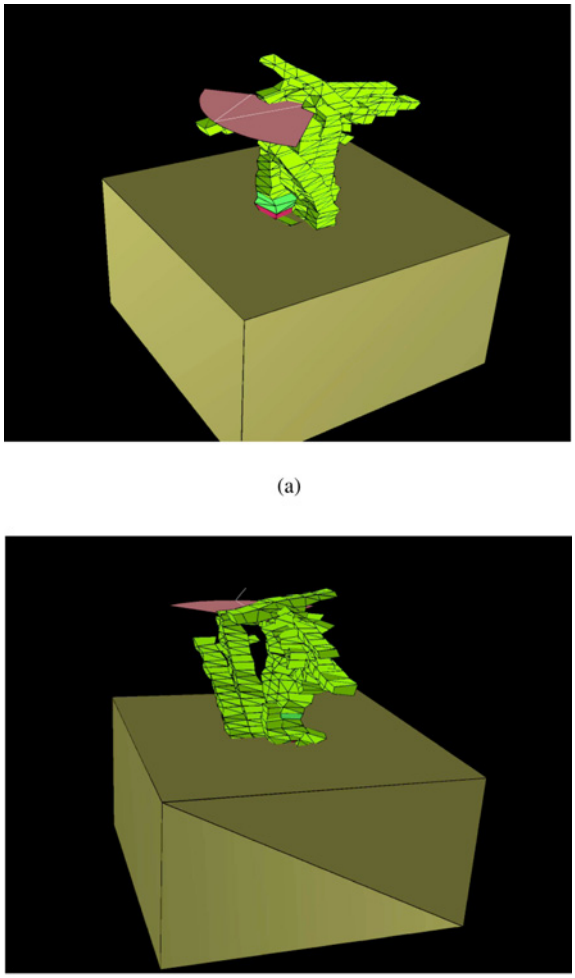


Fig. 20. (color online) Two views of a first phenotype (fitness value = 2.11), which is able to support the load, as indicated by the predominantly green color of the cells.

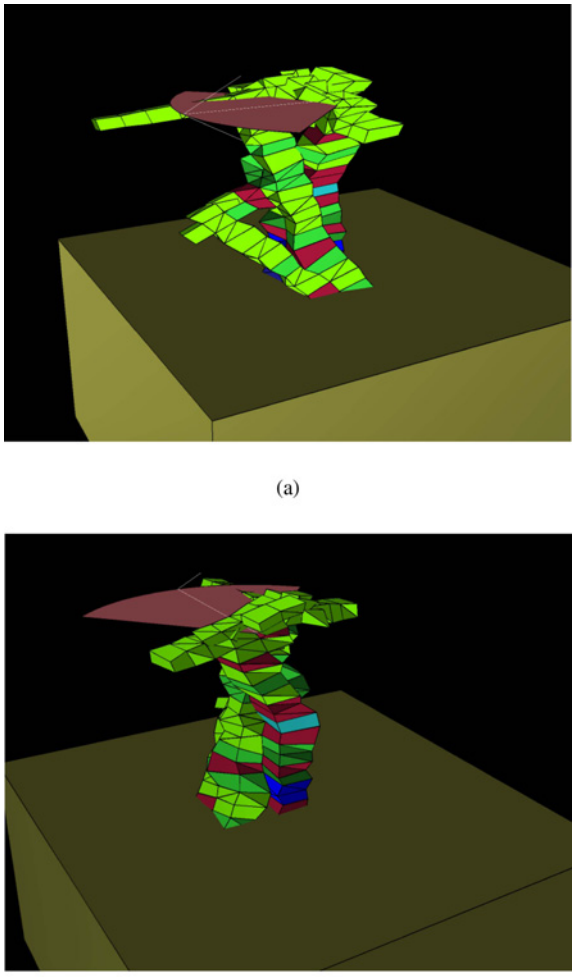


Fig. 21. (color online) Two views of a second phenotype (fitness value = 2.07), displaying a double-helix configuration. This phenotype is also able to support the load.

Beginning with a single cell attached to the ground, the evolved set of rules guides the growth and development of the phenotype, as described below.

a) Fig. 22(a): Initially, cell division occurs along the axis of the original cell facing toward the load morphogen, creating a vertical stack of three cells. Then the top cell in the stack divides laterally.
b) Fig. 22(b): Next, the lateral cell divides normal to the surface of the cell facing toward and away from the load morphogen. This creates the starting point for the two principal modular elements of the double helix of the structure.
c) Fig. 22(c): Growth of both elements continues. The third cell from the top of the right-hand element divides laterally, creating the starting point for the third modular element.
d) Fig. 22(d): Growth of all three modular elements continues, approaching the load morphogen. Note that all cells are colored green, indicating that they are lightly stressed. A fourth modular element begins to grow diagonally into the ground near the left-hand side of the base of the structure.
e) Fig. 22(e): Growth of the composite structure continues and reaches the load morphogen, inducing load onto the structure, as reflected in the red color of many of the cells. The bright red color indicates cells that are carrying stress above the yield stress of the material. The second and third modular elements have merged (through the action of cell adhesion). The fourth modular element continues to grow diagonally upwardly, and can be seen as the green cells extending to the right of the structure, near its base.
f) Fig. 22(f): Additional growth, and adhesion of cells near the top of the two principal modular elements, results in reduction of the stress in all of the cells to levels below the yield stress of the material.

Fig. 23 shows four different phenotypes which have been developed from the same genotype. The first three have similar fitness values and similar topology with some degree of variation. The fourth phenotype has different topology, since

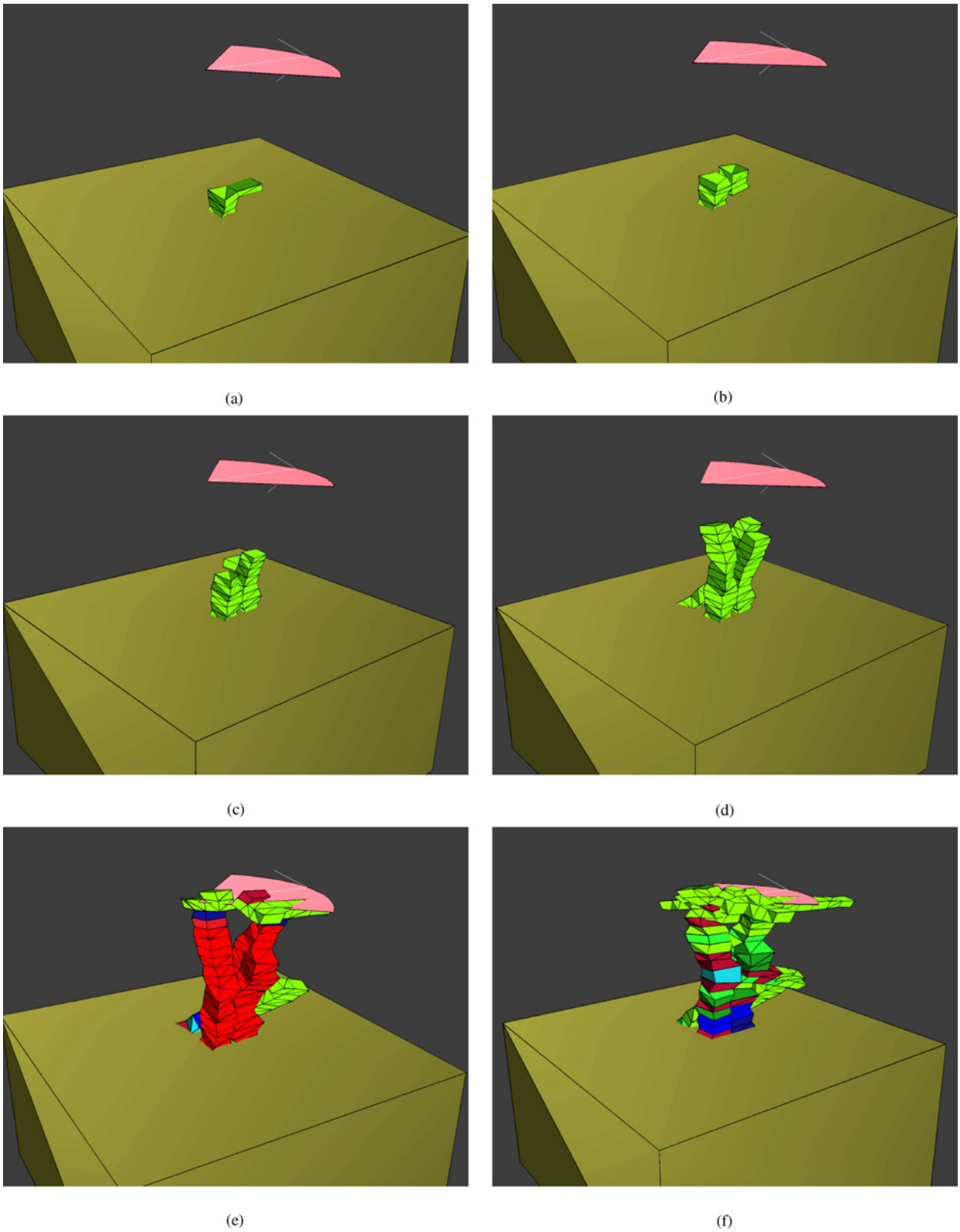


Fig. 22. (color online) Six stages in the growth and development of the phenotype shown in Fig. 21.

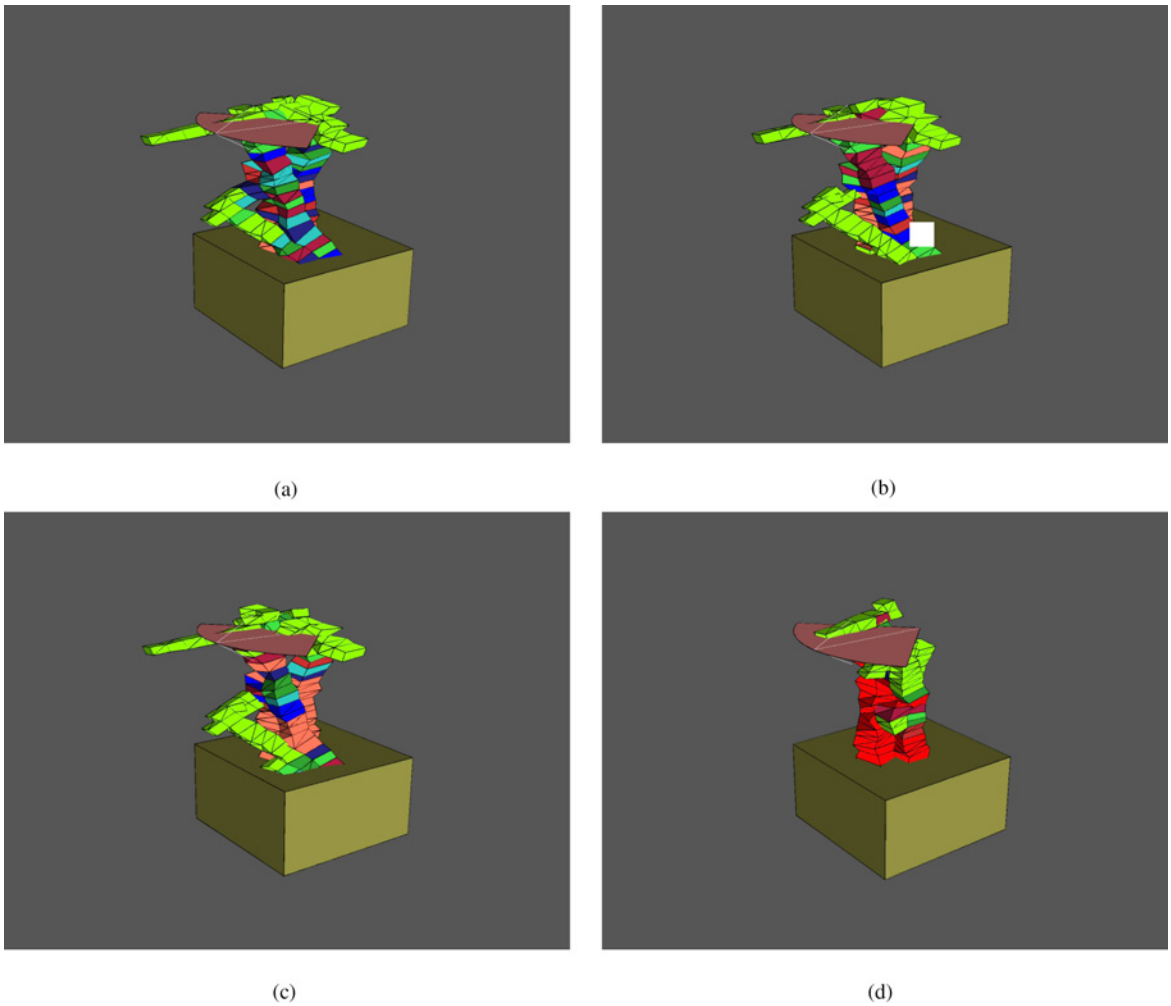


Fig. 23. (color online) Four different phenotypes, growing from the same genome. The topologies of the phenotypes in (a)–(c) are almost identical, aside from small variations which result from the environment. The phenotype in (d) has a visibly distinct topology.

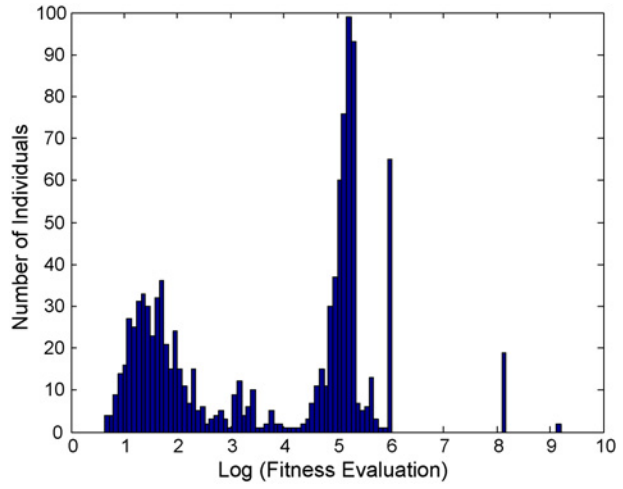


Fig. 24. Distribution of fitness values of phenotypes grown and developed from a single genotype.

it has not completely developed. Most of the cells are overstressed, which results in a higher fitness evaluation (and therefore lower performance).

Fig. 24 shows the fitness evaluations of hundreds of phenotypes which were developed from the same genotype. Two peaks of the fitness values can be observed, with nearby distributions similar to a Gaussian distribution. Phenotypes with fitness values near the lower peak have topology similar to the phenotypes in Fig. 23(a)–(c). The second group corresponds to phenotypes that perform more poorly (and hence have a higher fitness value) and are similar to the phenotype shown in Fig. 23(d). Due to variability in the environment, some phenotypes cannot complete their development process properly at some time step. The phenotypes which are able to continue developing after this time step usually continue to develop into successful mature phenotypes similar to those shown in Fig. 23(a)–(c).

4) *Genome Analysis:* The genome that gives rise to the phenotype depicted in Figs. 21 and 22 is shown in Fig. 25. There are seven different words in the genome with one word repeated 1113 times. This word contains four major characteristics: veto rules, anisotropic growth, cell division, and cell death. The veto rules suppress the execution of this word when the phenotype reaches a certain age, which means that this word is executed frequently at the beginning of the

```
R1Z0S3hC7aV95gB9b6c6hC5gC0cB1h9b6cB2c4b0gDidiS5cDi
R1113Z2W158tV72gB3b2c3bDiiiK
R1Z2W110bK
R1Z2A1dC0gB1g2c8gV72gB3b2c3bDiii
R1Z0C5hV95gDifS2cC2hK
R2Z1W5tV102aS5a
R1Z1W135gKC9g
```

Fig. 25. Genome for the phenotype shown in Figs. 21 and 22 contains seven different words.

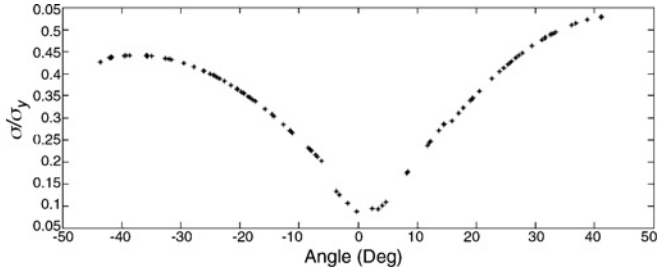


Fig. 26. Maximum mechanical stress, normalized to the material yield stress (σ/σ_y) versus angle of the load vector.

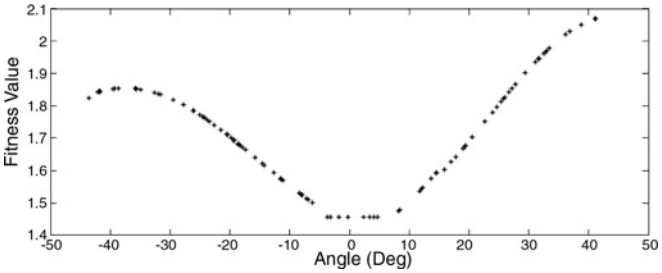


Fig. 27. Fitness value versus angle of the load vector.

developmental stage and produces a large number of cells. The cell death rule separates a group of cells and promotes the creation of two struts in the phenotype. The shearing operation turns the struts into two helices which wrap around each other.

The mechanism of rapid cell production at the initial stage of development has also been observed in nature. Embryo cells divide much more frequently at the beginning of development than at maturity. The repetition of rules in the genome may be one of the factors that produces a degree of modularity in the phenotype [1].

A. Additional Runs

One example of a phenotype produced by the combined processes of evolution and development which exhibits characteristics of modularity is presented above. Additional evolutions synthesized new phenotypes. One of these phenotypes is shown in Fig. 28. Even though the presented phenotype appears to be much less modular than the previous one, it has the capability to support the load, yet remains relatively light weight. This phenotype has a narrow element in the

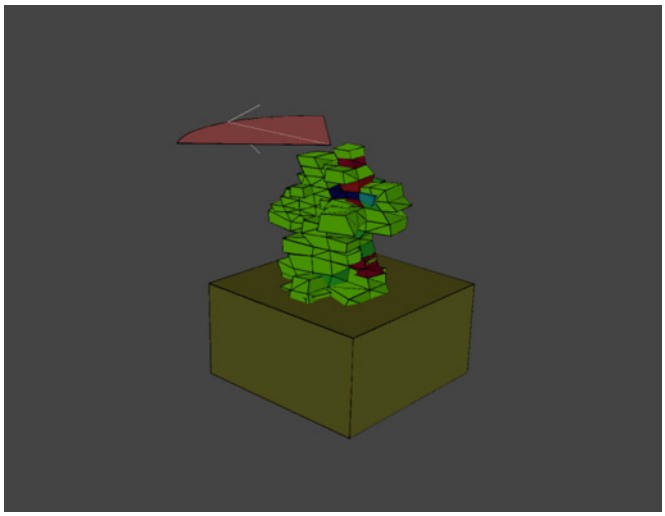


Fig. 28. (color online) Evolved phenotype which has been generated by additional simulation.

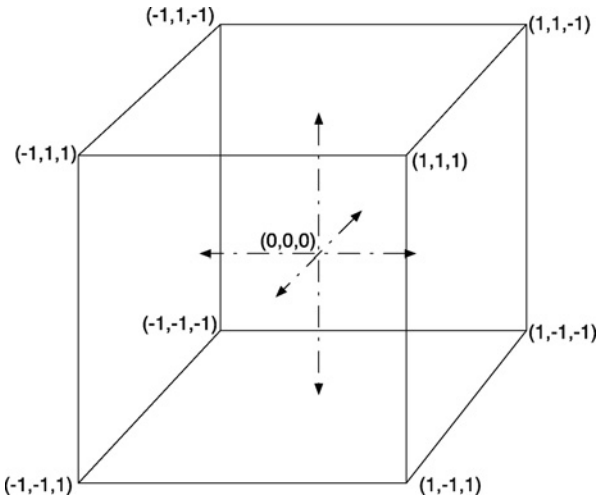


Fig. 29. Local–global mapping.

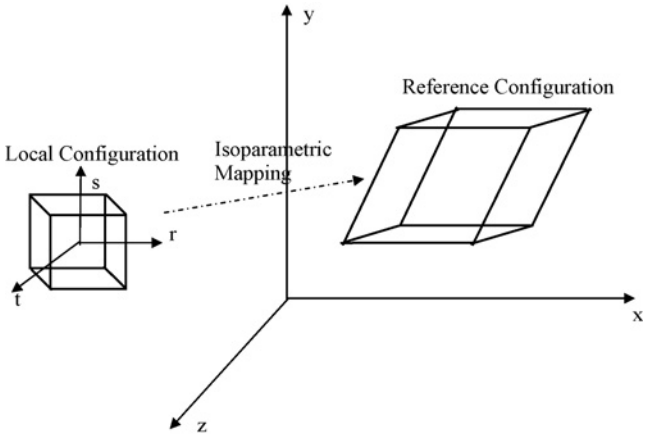


Fig. 30. Basic structural hexahedral "brick" 3-D finite element.

main direction of the load. This element has two effects: it increases the strength of the phenotype in the main direction of the load and it is relatively light. The base element in the phenotype is rather wide, which dramatically reduces the bending stresses on the base of the phenotype, where the bending stresses are largest. This result demonstrates that there are many alternative solutions to this type of problem. Each run of the evolutionary process synthesizes a new solution.

B. Constraints

As in practical engineering, where constraints play a major role during the design process, several physical constraints are active here, including thermodynamics, cell intersections, etc. An internal constraint to not grow ill-conditioned mesh configurations is imposed through the disease mechanism, which repairs or eliminates solutions with high numerical error. Additional constraints could be incorporated in the fitness function, for example phenotypes that exceed predetermined geometric limits can be given low fitness values. The advantage of not including such external constraints is that the evolutionary process is able to freely explore the design space, and synthesize solutions that provide high levels of performance without being constrained to a predetermined configuration.

C. Robustness

Prior results have shown that including expected variations in the evaluation of the fitness of individuals during evolution produces phenotypes that are robust to the variations [55]. In order to evaluate the robustness of the phenotype shown in Fig. 21, two plots are presented. Fig. 26 shows the maximum stress in the phenotype (normalized to the yield stress of the material) as a function of the angle of the load vector. Fig. 27 shows the fitness value of the phenotype as function of the angle of the load vector. The angle of the load vector at the center of the pink sector in Fig. 21 corresponds to the value 0. As can be seen in the data, the lowest fitness value (1.456, which corresponds to the highest performance) is achieved when the load angle is zero, and increases in a generally symmetrical pattern between ± 45 up to a fitness value of 2.07. This corresponds to a 30% change over the entire range of load vector angles, which indicates robust ability of the phenotype to successfully support the load over the full range of variation in the angle of the load vector.

XVI. Conclusion

The research reported in this paper has demonstrated that by simulating the interdependent processes of evolution and development, robust high performance design configurations can be synthesized *in silico* purely in response to engineering criteria. A simple set of *if*-conditional *then*-action rule elements can be evolved (in concert with the development of each evolved rule set into a phenotype, and the evaluation of the performance of each phenotype) into sets of rules (genotypes) that will develop into phenotypes (by executing the rules) that meet stated performance criteria. The results shown in this paper have demonstrated that the interplay of evolution and development can produce useful results, when guided only by evaluations of how well phenotypes meet the performance criteria.

Interestingly, some of the configurations synthesized by this method exhibit characteristics of modularity (distinct structural elements). These characteristics emerged spontaneously, without any pre-established preference or requirement.

XVII. Future work

Future work will include studying the structure of the evolved genomes and the configurations of the phenotypes they produce, focusing on the relationships between evolution and development and the rules and the modularity they produce, with the goal of better understanding the relationship between the evolved genome and the form of the phenotype. Further, the value of additional communication mechanisms between cells (similar to protein diffusion) will be examined. The effects of particular rules/genes on the phenotype during development will also be studied. Such an investigation into regulatory mechanisms which control the execution of rules during the developmental process may lead to a better understanding of the emergence of performance, robustness, and other characteristics observed in the synthesized phenotypes.

Appendix A
Numerical Schemes

The following sections provide a detailed explanation regarding the numerical methods and schemes used in this paper.

A. Configuration

Three types of configurations are defined: local, reference, and deformed. These configurations are useful for the computing process of certain parameters such as mechanical stresses, morphogen gradient, etc. The deformed configuration corresponds to the phenotype after it has been subjected to the load, and is thus deformed from its original state. This deformation generates strains which generate mechanical stresses. The reference configuration corresponds to the configuration of the phenotype before it has been subjected to the load. The local configuration is a topological configuration where every cell in the reference configuration is mapped to cube, centered at the origin, shown in Fig. 29.

B. Finite Element Scheme

Every cell in the structure is a 3-D hexahedral finite element. Every cell in the reference configuration has eight nodes. Every node has three coordinates as shown in Fig. 29. If the cell is topologically convex, then there exists a one-to-one isoparametric mapping which maps every point inside the cell in the reference configuration into a point inside a cube

centered on the origin with unit volume. A point inside the cell in its reference configuration is marked with X, Y, Z. The coordinates of the cell nodes in the reference configuration are defined as $X_i, Y_i, Z_i \quad i = 0, \ldots, 7$. A point inside the cell in its local configuration is marked with r, s, t. The mapping between a point in the local configuration of the cell (r, s, t) to a point in the reference configuration (X, Y, Z) is defined by (A.1). The shape function, N_i, is defined in (A.2)

$$\begin{aligned} X &= \sum_{i=0}^{7} N_i(r,s,t)\, X_i \\ Y &= \sum_{i=0}^{7} N_i(r,s,t)\, Y_i \\ Z &= \sum_{i=0}^{7} N_i(r,s,t)\, Z_i \end{aligned} \tag{A.1}$$

where

$$\begin{aligned} N_0 &= \tfrac{1}{8}(1+r)(1-s)(1+t) \\ N_1 &= \tfrac{1}{8}(1+r)(1+s)(1+t) \\ N_2 &= \tfrac{1}{8}(1-r)(1+s)(1+t) \\ N_3 &= \tfrac{1}{8}(1-r)(1-s)(1+t) \\ N_4 &= \tfrac{1}{8}(1+r)(1-s)(1-t) \\ N_5 &= \tfrac{1}{8}(1+r)(1+s)(1-t) \\ N_6 &= \tfrac{1}{8}(1-r)(1+s)(1-t) \\ N_7 &= \tfrac{1}{8}(1-r)(1-s)(1-t). \end{aligned} \tag{A.2}$$

A phenotype is simply a collection of cells in the reference configuration. The finite element approximation relates the displacement field $\mathbf{u}$ at every point inside the cell to the displacement field of its nodes $\mathbf{u_i}$ through the shape function in (A.2) according to

$$\mathbf{u} \approx \sum_{i=0}^{7} N_i(r,s,t)\mathbf{u}_i. \tag{A.3}$$

From linear elasticity theory the strain tensor ε at every point is defined by (A.4), where $\mathbf{L}$ is a linear operator defined in (A.5)

$$\varepsilon = \mathbf{L}\mathbf{u} \tag{A.4}$$

$$\mathbf{L} = \begin{bmatrix} \partial/\partial X & 0 & 0 \\ 0 & \partial/\partial Y & 0 \\ 0 & 0 & \partial/\partial Z \\ \partial/\partial Y & \partial/\partial Z & 0 \\ 0 & \partial/\partial Z & \partial/\partial Y \\ \partial/\partial Z & 0 & \partial/\partial X \end{bmatrix}. \tag{A.5}$$

Using both (A.4) and the linear operator $\mathbf{L}$, the *strain-nodal displacement matrix* $\mathbf{B}$ is defined as

$$\mathbf{B} = \mathbf{L}\mathbf{N} = \begin{bmatrix} \sum_{i=0}^{7} \frac{\partial N_i(r,s,t)}{\partial x} & 0 & 0 \\ 0 & \sum_{i=0}^{7} \frac{\partial N_i(r,s,t)}{\partial y} & 0 \\ 0 & 0 & \sum_{i=0}^{7} \frac{\partial N_i(r,s,t)}{\partial z} \\ \sum_{i=0}^{7} \frac{\partial N_i(r,s,t)}{\partial y} & \sum_{i=0}^{7} \frac{\partial N_i(r,s,t)}{\partial z} & 0 \\ 0 & \sum_{i=0}^{7} \frac{\partial N_i(r,s,t)}{\partial z} & \sum_{i=0}^{7} \frac{\partial N_i(r,s,t)}{\partial y} \\ \sum_{i=0}^{7} \frac{\partial N_i(r,s,t)}{\partial z} & 0 & \sum_{i=0}^{7} \frac{\partial N_i(r,s,t)}{\partial x} \end{bmatrix}. \tag{A.6}$$

The derivatives of the shape function A.6, with respect to the local coordinates, are mapped to their derivatives with respect to the reference configuration using the Jacobian, shown in

$$\mathbf{J} = \begin{bmatrix} \sum_{j=0}^{7} \frac{\partial N_j}{\partial r} x_j & \sum_{j=0}^{7} \frac{\partial N_j}{\partial r} y_j & \sum_{j=0}^{7} \frac{\partial N_j}{\partial r} z_j \\ \sum_{j=0}^{7} \frac{\partial N_j}{\partial s} x_j & \sum_{j=0}^{7} \frac{\partial N_j}{\partial s} y_j & \sum_{j=0}^{7} \frac{\partial N_j}{\partial s} z_j \\ \sum_{j=0}^{7} \frac{\partial N_j}{\partial t} x_j & \sum_{j=0}^{7} \frac{\partial N_j}{\partial t} y_j & \sum_{j=0}^{7} \frac{\partial N_j}{\partial t} z_j \end{bmatrix}. \tag{A.7}$$

Using the definition in (A.7), the derivatives of the shape function can be expressed with respect to the reference configuration as

$$\begin{bmatrix} \frac{\partial N_i}{\partial X} \\ \frac{\partial N_i}{\partial Y} \\ \frac{\partial N_i}{\partial Z} \end{bmatrix} = \mathbf{J}^{-1} \begin{bmatrix} \frac{\partial N_i}{\partial r} \\ \frac{\partial N_i}{\partial s} \\ \frac{\partial N_i}{\partial t} \end{bmatrix}. \tag{A.8}$$

In order to derive the relation between the external forces and the strains, the functional Π is defined, as shown in (A.9). The functional is defined as the difference between the internal strain energy and work on the boundary

$$\Pi = \int_{V^e} \sigma\varepsilon dV - \int_{V^e} \mathbf{u}\mathbf{b} dV - \int_{S^e} \mathbf{u}\mathbf{s} dS - \sum_{i=0}^{7} \mathbf{u}\mathbf{f}_i \tag{A.9}$$

where $\mathbf{b}$ represents the body force (e.g., gravity), σ is the mechanical stress tensor, $\mathbf{s}$ are the surface tractions, and $\mathbf{f_i}$ are the forces applied to the nodes.

The principle of virtual work applied to (A.9) results in

$$\begin{aligned} \delta\Pi \quad = \quad & \textstyle\int_{V^e} \sigma\delta(\varepsilon)\, dV - \int_{V^e} \delta(\mathbf{u})\, \mathbf{b} dV - \\ & \textstyle\int_{S^e} \delta(\mathbf{u})\, \mathbf{b} dS - \sum \delta(\mathbf{u})\, \mathbf{f}_p \quad = \quad 0 \end{aligned} \tag{A.10}$$

where

$$\begin{aligned} \textstyle\int_{V^e} \sigma\delta(\varepsilon)\, dV \quad = \quad & \textstyle\int_{V^e} \delta(\mathbf{u})\, \mathbf{b} dV + \\ & \textstyle\int_{S^e} \delta(\mathbf{u})\, \mathbf{s} dS + \sum \delta(\mathbf{u})\, \mathbf{f}_p. \end{aligned} \tag{A.11}$$

Equation (A.10) represents the relationship between the external forces, body forces, and surface traction, to the stress and the strains at every point in the cell. Two assumptions are made; the strain-displacement relation is linear, as shown in (A.4), and the constitutive equation which relates stress to strain is linear as well. The constitutive relation is presented in (A.12), where $\mathbf{D}$ is the material property matrix defined in (A.13). The material properties E and ν are the elastic modulus and Poisson's ratio, respectively

$$\sigma = \mathbf{D}\varepsilon \tag{A.12}$$

$$\mathbf{D} = \frac{1}{E}\begin{bmatrix} 1 & -\nu & -\nu & & & & \\ -\nu & 1 & -\nu & & & & \\ -\nu & -\nu & 1 & & & & \\ & & & 1+\nu & & & \\ & & & & 1+\nu & & \\ & & & & & 1+\nu & \end{bmatrix}. \tag{A.13}$$

Using (A.4), (A.12), and (A.6) in (A.10), a set of eight linear equations are obtained, represented in matrix form as

$$\mathbf{K}\mathbf{u}_i^e = \mathbf{f}_i^e \tag{A.14}$$

where $\mathbf{K}$ is the stiffness matrix, defined in (A.15). Equation (A.15) is defined as the integration of $\mathbf{B}^T\mathbf{D}\mathbf{B}$ over the volume of the cell. The stiffness matrix should be positive definite and symmetric

$$\mathbf{K} = \int_{V^e} \mathbf{B}^T\mathbf{D}\mathbf{B}dV. \tag{A.15}$$

1) *Integration Scheme:* The stiffness matrix in (A.15) requires integration across the volume of the cell. In order to perform this integration efficiently, the Gauss integration method is used. The expression $\mathbf{B}^T\mathbf{D}\mathbf{B}dV$ in (A.15) is a function of the local coordinates r, s, t only. The first step is to write the integrand in (A.15) over a cube in the local coordinate system. Using the definition of the Jacobian in (A.7), the stiffness matrix can be written as

$$K^e = \int_{-1}^{1}\int_{-1}^{1}\int_{-1}^{1} \mathbf{B^T D B}\,|\det \mathbf{J}|\,drdsdt. \tag{A.16}$$

The integral in (A.16) can be computed numerically using the Gauss integration method, as shown in (A.17). The quadrature points r_i, s_i, t_i are defined in (A.18)

$$K^e = \sum_{r_i,s_i,t_i\ i=1}^{8} |\det \mathbf{J}|\left(\mathbf{B^T D B}\right)\Big|_{r_i,s_i,t_i} \tag{A.17}$$

$$r_i, s_i, t_i = \left\{\pm\sqrt{1/3}, \pm\sqrt{1/3}, \pm\sqrt{1/3}\right\}. \tag{A.18}$$

2) *Assembly Scheme:* The system of equations defined in (A.14) corresponds to a single cell. In order to evaluate the displacements of the entire structure, all of the local stiffness matrices must be collected into a single global stiffness matrix $\bar{\mathbf{K}}$. The corresponding external forces, body forces, and surface tractions from all nodes must also be collected into a global vector $\bar{\mathbf{f}}$. At the end of the assembly routine a system of linear equations is produced, shown in

$$\bar{\mathbf{K}}\bar{\mathbf{u}} = \bar{\mathbf{f}}. \tag{A.19}$$

The first step is to assign sequential numbers to all the nodes. Every node is assigned three numbers, since the displacement field is a 3-D vector. A typical system of local equations in the form of (A.12) is presented in

$$\begin{bmatrix} K_{00} & & K_{07} \\ & . & \\ K_{70} & & K_{77} \end{bmatrix}\begin{bmatrix} u_{p_0} \\ . \\ u_{p_7} \end{bmatrix} = \begin{bmatrix} f_{p_0} \\ . \\ f_{p_7} \end{bmatrix}. \tag{A.20}$$

The global indices of the nodes are defined as p_0 to p_7. Starting with an empty global stiffness matrix, every row in the local stiffness matrix will be inserted into the global stiffness matrix in the following way; the index of the node which corresponds to this row will be the number of rows in the global stiffness matrix. Every term is then sequentially placed in column indices according to p_0 to p_7.

The assembly process for the external force will simply assign the force on a node to its location in the global matrix. This scheme will eventually generate a global set of equations for which the displacement field at each node can be computed. Once the displacement field is computed, the mechanical stress at each point inside the cell is determined using (A.4) and (A.12).

Appendix B
Cell Adhesion Scheme

The following sections provide an overview of the scheme used here for adhesion between neighboring cells.

A. Cell Intersection

Two cells may intersect during their growth process. The location of the center points of the cells is stored in a sorted array. After every execution of any geometric gene, a test for intersection of cells is performed. The intersection test is performed in two steps. In the first step, a cell is tested against eight or fewer adjacent cells which are ordered in the sorted array. Every cell is treated as a sphere with a radius, which is the maximum distance between any of the nodes of the cell and its center. If two spheres intersect then a more accurate test is performed using the enclosed ellipsoid algorithm [56]. The adhesion process will be applied to any two intersecting cells.

B. Face Determination

Cells in nature, prior to adhesion, communicate by way of a chemical signaling mechanism [32]. This process is mimicked here by considering each finite element to be an analog of a

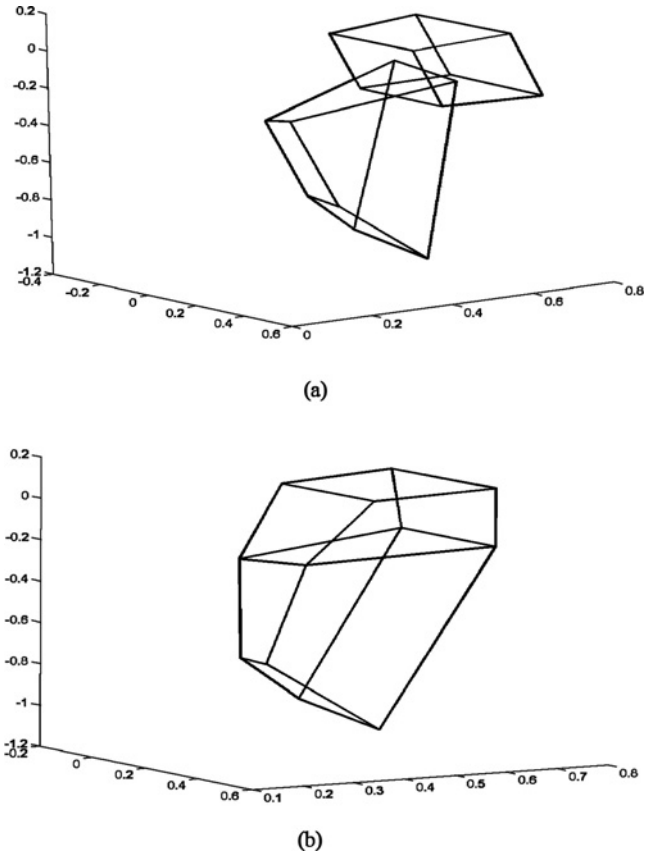


Fig. 31. Example adhesion. (a) Two intersecting cells prior to adhesion. (b) Two cells subsequent to adhesion.

cell. Each cell generates a signal at its center and this signal propagates through the faces of the cell and into the adjacent cells, schematically illustrated in Fig. 5(a). Each signal is characterized by an exponential function, shown in (B.1). The signal decays exponentially from the center of the cell

$$\mathbf{S} = \exp(-\alpha \mathbf{d}) \tag{B.1}$$

where α is a constant, $\mathbf{d}$ is the vector of the coordinates of the center of the cell, and $\mathbf{S}$ is a gradient vector.

The face of each cell which experiences the maximum intensity of the signal from the adjacent cell is chosen for adhesion, as illustrated schematically in Fig. 5(b).

C. Node Determination

Every face contains four nodes. The adhesion process merges pairs of nodes, one from each face, into a new node located at the average coordinates of the two nodes prior to adhesion. Utilizing this merging process results in 24 possible combinations of nodes for adhesion. In the approach presented here, the combination of nodes which results in a minimum distortion for the two cells and thus the minimum strain energy is selected.

D. Strain Energy Minimization

The initial assumption for the location of new nodes is the average of their initial locations, however, this assumption cannot guarantee a global minimization in terms of strain energy. An additional energy minimization is performed to achieve a global minimum in terms of strain energy using the conjugate gradient method.

An example of an adhesion of two cells is shown in Fig. 31.

ACKNOWLEDGMENTS

The authors gratefully acknowledge the generosity of Prof. S. Krishnan at the California Institute of Technology in making his computing cluster available for their evolutionary embryogeny computations. The computations reported in this paper were also carried out, in part, at the Jet Propulsion Laboratory. The authors would also like to thank Prof. R. Phillips and Prof. C. Adami for their discussion and ideas that assisted in guiding this research.

REFERENCES

[1] S. B. Carroll, *Endless Forms Most Beautiful: The New Science of Evo Devo and the Making of the Animal Kingdom*. New York: Norton, 2005, p. 371.
[2] A. R. Palmer, "Symmetry breaking and the evolution of development," *Science*, vol. 306, no. 5697, pp. 828–833, Oct. 2004.
[3] K. Sterelny and P. Kitcher, "The return of the gene," *J. Philos.*, vol. 85, no. 7, pp. 339–361, Jul. 1988.
[4] A. Stoltzfus, "Mutationism and the dual causation of evolutionary change," *Evol. Develop.*, vol. 8, no. 3, pp. 304–317, May–Jun. 2006.
[5] S. B. Carroll, J. K. Grenier, and S. D. Weatherbee, *From DNA to Diversity: Molecular Genetics and the Evolution of Animal Design*, 2nd ed. Malden, MA: Blackwell, 2004.
[6] P. J. Bentley and S. Kumar, "Three ways to grow designs: A comparison of embryogenies for an evolutionary design problem," in *Proc. Genetic Evol. Comput. Conf.*, Orlando, FL: Morgan Kaufmann, 1999, pp. 35–43.
[7] M. P. Bendsøe and O. Sigmund, *Topology Optimization: Theory, Methods and Applications*, 2nd ed. Berlin, Germany: Springer-Verlag, 2003.
[8] H. Lipson and J. B. Pollack, "Automatic design and manufacture of robotic lifeforms," *Nature*, vol. 406, no. 6799, pp. 974–978, Aug. 2000.
[9] J. B. Pollack, H. Lipson, G. Hornby, and P. Funes, "Three generations of automatically designed robots," *Artif. Life*, vol. 7, no. 3, pp. 215–223, Summer 2001.
[10] V. Zykov, E. Mytilinaios, B. Adams, and H. Lipson, "Self-reproducing machines," *Nature*, vol. 435, no. 7038, pp. 163–164, May 2005.
[11] J. H. Holland, *Adaptation in Natural and Artificial Systems*. Ann Arbor, MI: University of Michigan Press, 1975.
[12] P. Eggenberger, "Evolving morphologies of simulated 3-D organisms based on differential gene expression," in *Proc. 4th Eur. Conf. Artif. Life*, Cambridge, MA: MIT Press, 1997, pp. 205–213.
[13] W. G. Macready, A. G. Siapas, and S. A. Kauffman, "Criticality and parallelism in combinatorial optimization," *Science*, vol. 271, no. 5245, pp. 56–59, Jan. 1996.
[14] C. P. Bowers, "Evolving robust solutions with a computational embryogeny," in *Proc. U.K. Workshop Comput. Intell.*, Bristol, U.K.: University of Bristol, 2003, pp. 181–188.
[15] C. P. Bowers, "Simulating evolution with a computational model of embryogeny: Obtaining robustness from evolved individuals," in *Proc. 8th Eur. Conf. Artif. Life: Adv. Artif. Life (ECAL)*, 2005, pp. 149–158.
[16] S. Kumar and P. J. Bentley, "Computational embryology: Past, present and future," in *Advances in Evolutionary Computing, Theory and Applications*, A. Ghosh and S. Tsutsui, Eds. New York: Springer-Verlag, 2003, pp. 461–478.
[17] C. P. Bowers, "Formation of modules in a computational model of embryogeny," in *Proc. 2005 Congr. Evol. Comput. (CEC)*, vol. 1. Piscataway, NJ: IEEE Press, Sep. 2005, pp. 537–542.
[18] H. Lipson, J. B. Pollack, and N. P. Suh, "On the origin of modular variation," *Evolution*, vol. 56, no. 8, pp. 1549–1556, Aug. 2002.
[19] N. Kashtan and U. Alon, "Spontaneous evolution of modularity and network motifs," in *Proc. Nat. Academy Sci.*, vol. 102, no. 39. Sep. 2005, pp. 13773–13778.

[20] P. Maes, "Modeling adaptive autonomous agents," *Artificial Life*, vol. 1, no. 9, pp. 135–162, 1994.
[21] J. Vaario and K. Shimohara, "Synthesis of developmental and evolutionary modeling of adaptive autonomous agents," in *Proc. Int. Conf. Artif. Neural Netw.*, LNCS, 1997, pp. 721–726.
[22] J. Vaario, K. Hori, and S. Ohsuga, "Toward evolutionary design of autonomous systems," *Int. J. Comput. Simulation*, vol. 5, no. 2, pp. 187–206, 1994.
[23] J. Vaario, *The Role of Environmental Adaptation in Evolutionary Computation*, Aug. 1993 [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.51.9777
[24] H. Kitano, "Designing neural networks using genetic algorithms with graph generation system," *Complex Syst.*, vol. 4, no. 4, pp. 461–476, 1990.
[25] H. Kitano, "A simple model of neurogenesis and cell differentiation based on evolutionary large-scale chaos," *Artif. Life*, vol. 2, no. 1, pp. 79–99, Jan. 1995.
[26] T. Kowaliw, P. Grogono, and N. Kharma, "Environment as a spatial constraint on the growth of structural form," in *Proc. 9th Annu. Conf. Genetic Evol. Comput. (GECCO)*, New York: ACM, 2007, pp. 1037–1044.
[27] T. Kowaliw, P. Grogono, and N. Kharma, "The evolution of structural design through artificial embryogeny," in *Proc. IEEE Symp. Artif. Life (ALIFE)*, Apr. 2007, pp. 425–432.
[28] T. Steiner, Y. Jin, and B. Sendhoff, "A cellular model for the evolutionary development of lightweight material with an inner structure," in *Proc. 10th Annu. Conf. Genetic Evol. Comput. (GECCO)*, New York: Association for Computing Machinery, 2008, pp. 851–858.
[29] P. Funes and J. Pollack, "Computer evolution of buildable objects," in *Proc. 4th Eur. Conf. Artif. Life*, Cambridge, MA: MIT Press, 1997, pp. 358–367.
[30] F. B. Prinz and L. E. Weiss, "Novel applications and implementations of shape deposition manufacturing," *Naval Res. Rev.*, vol. 50, no. 3, pp. 19–26, 1998.
[31] B. Sendhoff, "Evolutionary optimised ontogenetic neural networks with incremental problem complexity during development," in *Proc. 2000 Congr. Evol. Comput.*, vol. 2. Piscataway, NJ: IEEE, Jul. 2000, pp. 1443–1450.
[32] B. Alberts, A. Johnson, J. Lewis, M. Raff, K. Roberts, and P. Walter, *Molecular Biology of the Cell*, 4th ed. New York: Garland Publishing, 2002.
[33] C. J. Sherr, "Principles of tumor suppression," *Cell*, vol. 116, pp. 235–246, Jan. 2004.
[34] O. C. Zienkiewicz and R. L. Taylor, *The Finite Element Method*, 6th ed. Oxford, U.K.: Elsevier Butterworth-Heinemann, 2005.
[35] P. Puech, A. Taubenberger, F. Ulrich, M. Krieg, D. J. Muller, and C. Heisenberg, "Measuring cell adhesion forces of primary gastrulating cells from zebrafish using atomic force microscopy," *J. Cell Sci.*, vol. 118, pp. 4199–4206, Sep. 2005.
[36] W. S. Tan and Y. L. Chen, "Quantitative investigations of cell-bubble interactions using a foam fractionation technique," *Cytotechnology*, vol. 15, nos. 1–3, pp. 321–328, Feb. 1994.
[37] E. Coen, A. Rolland-Lagan, M. Matthews, J. A. Bangham, and P. Prusinkiewicz, "The genetics of geometry," in *Proc. Nat. Academy Sci.*, vol. 101, no. 14, Apr. 2004, pp. 4728–4735.
[38] Y. Zhang, C. Bajaj, and B.-S. Sohn, "3-D finite element meshing from imaging data," *Comput. Methods Appl. Mech. Eng.*, vol. 194, pp. 5083–5106, Nov. 2005.
[39] C. B. W. Pedersen, T. Buhl, and O. Sigmund, "Topology synthesis of large-displacement compliant mechanisms," *Int. J. Numerical Methods Eng.*, vol. 50, pp. 2683–2705, Feb. 2001.
[40] L. Hufnagel, A. A. Teleman, H. Rouault, S. M. Cohen, and B. L. Shraiman, "On the mechanism of wing size determination in fly development," in *Proc. Nat. Academy Sciences*, vol. 104, no. 10, Mar. 2007, pp. 3835–3840.
[41] X. Wertz, D. Schoëvaërt, H. Maitournam, P. Chassignet, and L. Schwartz, "The effect of hormones on bone growth is mediated through mechanical stress," *Comptes Rendus Biol.*, vol. 329, no. 2, pp. 79–85, Feb. 2006.
[42] B. I. Shraiman, "Mechanical feedback as a possible regulator of tissue growth," in *Proc. Nat. Academy Sciences*, vol. 102, no. 9, Mar. 2005, pp. 3318–3323.
[43] C. Day, "Mechanical force may determine the final size of tissues," *Phys. Today*, vol. 60, no. 4, pp. 20–21, Apr. 2007.
[44] G. B. West, J. H. Brown, and B. J. Enquist, "Growth models based on first principles or phenomenology?" *Functional Ecol.*, vol. 18, no. 2, pp. 188–196, Apr. 2004.
[45] M. Kleiber, "Body size and metabolism," *Hilgardia*, vol. 6, pp. 315–353, 1932.
[46] M. Kleiber, "Body size and metabolic rate," *Physiol. Rev.*, vol. 27, no. 4, pp. 511–541, Oct. 1947.
[47] P. M. Knupp, "A method for hexahedral mesh shape optimization," *Int. J. Numerical Methods Eng.*, vol. 58, no. 2, pp. 319–332, Sep. 2003.
[48] L. A. Freitag and P. M. Knupp, "Tetrahedral mesh improvement via optimization of the element condition number," *Int. J. Numerical Methods Eng.*, vol. 53, no. 6, pp. 1377–1391, 2002.
[49] P. M. Knupp, "Hexahedral and tetrahedral mesh untangling," *Eng. Comput.*, vol. 17, no. 3, pp. 261–268, Oct. 2001.
[50] P. M. Knupp, "Achieving finite element mesh quality via optimization of the Jacobian matrix norm and associated quantities. Part II: A framework for volume mesh optimization and the condition number of the Jacobian matrix," *Int. J. Numerical Methods Eng.*, vol. 48, no. 8, pp. 1165–1185, Jul. 2000.
[51] P. M. Knupp, "Achieving finite element mesh quality via optimization of the Jacobian matrix norm and associated quantities. Part I: A framework for volume mesh optimization," *Int. J. Numerical Methods Eng.*, vol. 48, no. 3, pp. 401–420, May 2000.
[52] C. M. Fonseca and P. J. Fleming, "An overview of evolutionary algorithms in multiobjective optimization," *Evol. Comput.*, vol. 3, no. 1, pp. 1–16, Spring 1995.
[53] N. Srinivas and K. Deb, "Multiobjective optimization using nondominated sorting in genetic algorithms," *Evol. Comput.*, vol. 2, no. 3, pp. 221–248, Fall 1994.
[54] M. J. Scott and E. K. Antonsson, "Aggregation functions for engineering design trade-offs," *Fuzzy Sets Syst.*, vol. 99, no. 3, pp. 253–264, Nov. 1998.
[55] L. Ma and E. K. Antonsson, "Robust mask-layout and process synthesis," *J. Microelectomech. Syst.*, vol. 12, no. 4, p. 12, Aug. 2003.
[56] M.-Y. Ju, J.-S. Liu, S.-P. Shiang, Y.-R. Chien, K.-S. Hwang, and W.-C. Lee, "A novel collision detection method based on enclosed ellipsoid," in *Proc. IEEE Int. Conf. Robotics Autom.*, 2001, vol. 3, pp. 2897–2902.
[57] O. C. Zienkiewicz and R. L. Taylor, *Finite Element Method: Volume 2, Solid Mechanics*. Amsterdam, The Netherlands: Elsevier, 2000.
[58] P. J. Flory, "Thermodynamic relations for high elastic materials," *Trans. Faraday Society*, vol. 57, pp. 829–838, 1961.
[59] W. W. Hager and H. Zhang, "A conjugate gradient method with guaranteed descent and an efficient line search," *Soc. Ind. Appl. Math. J. Optimization*, vol. 16, no. 1, pp. 170–192, 2005.
[60] W. W. Hager and H. Zhang, "CG descent: A conjugate gradient method with guaranteed descent," *ACM Trans. Math. Softw. (TOMS)*, vol. 32, no. 1, pp. 113–137, Mar. 2006.
[61] R. Fletcher and C. M. Reeves, "Function minimization by conjugate gradients," *Comput. J.*, vol. 7, no. 2, pp. 149–154, 1964.
[62] J. Zhang and C. Xu, "Properties and numerical performance of quasi-Newton methods with modified quasi-Newton equations," *J. Comput. Appl. Math.*, vol. 137, no. 2, pp. 269–278, Dec. 2001.
[63] S. Kumar and P. J. Bentley, "Implicit evolvability: An investigation into the evolvability of an embrogeny," in *Proc. Genetic Evol. Comput. Conf.*, Las Vegas, NV: Morgan Kaufmann, 2000.
[64] S. Kumar and P. J. Bentley, *On Growth, Form and Computers*. New York: Academic Press, 2003.
[65] C. P. Bowers and J. A. Bullinaria, "Embryological modelling of the evolution of neural architecture," in *Modeling Language, Cognition and Action*, A. Cangelosi, G. Bugmann, and R. Borisyuk, Eds. Singapore: World Scientific, 2005, pp. 375–384.
[66] C. P. Bowers, "Simulating evolution with a computational model of embryogeny," Ph.D. dissertation, School Computer Science, Univ. Birmingham, Birmingham, U.K., 2006.

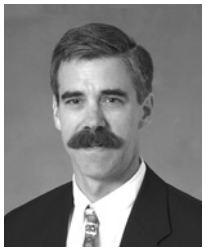
Or Yogev received the B.S. and M.S. degrees in mechanical engineering from the Technion Israel Institute of Technology, Haifa, Israel, in 2003 and 2005, respectively, and the Ph.D. degree in applied mechanics from the California Institute of Technology, Pasadena, in 2008.

He is currently employed as a Senior Research Scientist at eSolar, Inc., Pasadena, CA, and as a Visitor at the Engineering Design Research Laboratory, Division of Engineering and Applied Science, California Institute of Technology. His research interests include developing a novel design synthesis method and understanding basic processes in natural evolution, by mimicking embryogenesis and evolutionary processes in a computational model.

Andrew A. Shapiro (M'00–SM'09) received the B.S. degree in chemical engineering from the University of California, Berkeley, in 1981, the M.S. degree in materials science from the University of California, Los Angeles, in 1989, and the Ph.D. degree in materials science from the University of California, Irvine, in 1998.

He is currently an Associate Adjunct Professor in the Department of Electrical Engineering and Computer Science, University of California, Irvine. He holds positions in the materials engineering and manufacturing technology interdisciplinary concentration, and in the program in industrial ecology at the University of California, Irvine. He is also a Visiting Associate in mechanical engineering at the California Institute of Technology, Pasadena, where he is performing research in computational embryogeny, environmentally sustainable manufacturing of electronics, and optical and high frequency packaging. He is the Division Lead Technologist for the Enterprise Engineering Division, Jet Propulsion Laboratory, California Institute of Technology, where he has been working in microelectronic interconnects for more than 25 years. He is a Project Manager and a Principal Engineer at the Jet Propulsion Laboratory, where he is implementing new electronic, radio frequency, and optical technologies into space missions, and performing research on extreme environments, long-life spacecraft missions, and evolutionary design for reliability techniques. He has worked as a Member of the Technical Staff at the Rockwell International Corp., Manhasset, NY, and Hughes Aircraft, Culver City, CA, where he was responsible for the packaging of a number of phased array radars, and ran high density interconnect and low temperature cofired ceramics production lines. As a Principal Scientist at Broadcom (formerly NewPort Communications), Irvine, CA, he designed the first commercial polymer 10 GHz Si packages, as well as 10 and 40 GHz optoelectronic modules. He holds over nine patents, has presented at more than 70 conferences, and has authored numerous peer-reviewed publications and published proceedings.

Dr. Shapiro is currently an Associate Editor for the IEEE TRANSACTIONS ON PACKAGING MANUFACTURING. He has served on several national committees, including the National Environmental Methods Index Optoelectronics Roadmap, the Electronic Components and Technology Conference on Optoelectronics, the Electronics System-Integration Technology Conference on Optoelectronics, and the International Microelectronics and Packaging Society on Education.



Erik K. Antonsson (M'82) received the B.S. degree in mechanical engineering from Cornell University, Ithaca, NY, in 1976, and the M.S. and Ph.D. degrees in mechanical engineering from the Massachusetts Institute of Technology, Boston, in 1978 and 1982, respectively.

He is a Professor of Mechanical Engineering at the Division of Engineering and Applied Science, California Institute of Technology, Pasadena, where he organized the Engineering Design Research Laboratory, and where he has conducted research and taught since 1984. He previously served as the Executive Officer of the Department of Mechanical Engineering, California Institute of Technology. From September 2002 to January 2006, he was on leave from the California Institute of Technology and served as the Chief Technologist for the Jet Propulsion Laboratory of NASA. He is the author of more than 120 scholarly papers in the engineering design research literature, has edited three books, and holds eight U.S. patents.

He is a Fellow of the American Society of Mechanical Engineers and is currently on the editorial board of the international journal, *Research in Engineering Design*.